MICROBYTE

TODO COMPUTACION MAYO 1985 Nº 12 $ 150



Indice Analítico: Volumen 1

El Método PERT

Programando el 6502

Programas Atari, Sinclair, Commodore

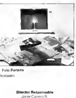


Foto Portada
Felicidades

Director Responsable
Jorge Carrera R
Coordinador General
José Kaltman T
Director Publicidad y RR PP
Ariel Leporati P
Publicidad
Juan Pablo Novoa
Ventas
Orlando Zepeda
Redaccion Periodistica
Myriam Pinto M
Directora de Arte
Paz Barba
Fotografia
González y Gutfloy
Cuerpo Editorial
Jaime Aravena
Jorge Cea
Carlos Contreras
Corresponsales en el exterior
Luis Kaltman T (Londres)
Alfredo Zarowsky (Paris)
Victor Kahan (Ohio)
Fotocomposición
Lase Ltda
Representante Legal
Jorge Carrera R
Direccion Merced 346 Of F
Fono 393866
Distribucion
Antartica S A
Impresion
Tamarcos quien solo actua como impresor

Microbyte es una publicación mensual de KVC Asociados

Colaboraciones de los lectores son bienvenidas y serán publicadas previa revisión con un pago de acuerdo a tipo de colaboración y calidad

Las colaboraciones deben venir tipeadas o impresas a doble espacio y, si es posible, acompañadas de material grafico. En el caso de listados de programas mayores de 15 lineas, es preferible enviar cassette o disco y una explicación de su contenido

SUBSCRIPCIONES
Valor subscripciones semestral (6 Ejs.)
Correo Certif Stgo y Prov $ 850
Entrega por mano Stgo $ 800
Valor subscripciones anual (12 Ejs.)
Correo Certif Stgo y Prov $ 1 600
Entrega por mano Stgo $ 1 500
Solicite un representante al fono 393866 en Merced N° 346, Of F Santiago Chile

Cuando hace un año en esta misma fecha salía a la calle el primer numero de nuestra revista, todos quienes trabajamos en ella teníamos claro que el éxito de ésta dependería no sólo de nuestro trabajo sino principalmente del grado de apoyo que recibiera de todos ustedes

Es por esto que al cumplir su primer año, Microbyte debe agradecer y felicitar a todos quienes han hecho posible que hoy esta revista sea una digna publicación técnica con un nivel de contenido y presentacion que nos llena de orgullo

Agradecemos a todas las empresas proveedoras de equipos y suministros que nos han acompañado durante este año y que a la vez están demostrando que también en Chile existe una infraestructura capaz de apoyar a un medio de la calidad de Microbyte

Agradecemos a todos nuestros lectores que mes a mes se han ido sumando a esta gran familia de Microbyte, estimulándonos con una nutrida correspondencia plena de inquietudes y sugerencias que nos ayuda a mejorar permanentemente el nivel y enfoque de los temas que tratamos

Agradecemos a todos quienes nos han enviado sus valiosas colaboraciones que nos permiten decir con propiedad que Microbyte es una revista de todos y para todos El rol de una publicacion como ésta, es precisamente servir como puente para que todos aquellos que pueden aportar con el producto de su conocimiento y experiencia al desarrollo de la disciplina informática en el país encuentren en sus páginas una tribuna

Al cumplir su primer año, Microbyte puede decirles que han cumplido con las metas y con creces En conjunto ustedes y nosotros hemos realizado con seriedad y entusiasmo una labor que le permite contar hoy al país con un medio de difusion, capacitacion, entretencion y punto de encuentro para todos los interesados en este apasionante y promisorio mundo de la informática

Desde el pie de esta página, en que nos hemos multiplicado para mejor cumplir con ustedes, les deseamos que en su segundo ano, Microbyte sea cada vez mejor y más vuestro



NOTICIAS

NOVEDADES

Fabricantes europeos se unen alrededor deUnix

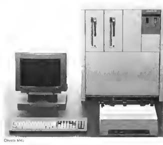
A partir de este año, seis de los principales fabricantes de computadores europeos (Bull, ICL, Nixdorf, Olivetti, Phillips y Siemens) comenzaran a trabajar en conjunto sobre un ambiente Unix similar, para así desarrollar un parque de equipos que hagan interesante producir software en Unix para ellos

De este trabajo, se pretende desarrollar una gama de lenguajes, compiladores y herramientas que faciliten el desarrollo de una base de software Unix para sus equipos

Hasta ahora, dada la diversidad de variantes de Unix que han sido implementadas en diversas marcas de equipos, han sido escasas las empresas productoras de software que han entrado al mercado del Unix

Este desarrollo a la vez de incentivar una mayor producción de software con una consiguiente disminución de precios para los usuarios, al mismo tiempo promete aumentar las capacidades de comunicación entre diferentes marcas de hardware

Sin embargo, aun existen serias dudas respecto al futuro de Unix debido principalmente a la actitud ambivalente de IBM,la cual hace dudar a muchos usuarios respecto a la conveniencia de migrar hacia esa sistema operativo



Olivetti M40

Nuevo Drive IBM para mainframes.

IBM introdujo al fin su largamente anunciado drive de "doble capacidad" para mainframes que literalmente duplica la capacidad del 3380 aumentando su precio en sólo un 52% El nuevo drive que viene en dos modelos, tendrá un precio estimado de US$ 110 000 y 134 000

En 1984, IBM vendió alrededor de 35 000 unidades del 3380, tres veces más que el total de todos sus competidores

El video disco desplaza a la microficha

La Biblioteca del Congreso en Estados Unidos está realizando un experimento, mediante el cual serán almacenados en videodisco más de un millón de páginas de texto, las que podrán ser accesadas por los usuarios en terminales especiales de video

Esta biblioteca que cuenta con más de 80 millones de ítems tenía el problema de accesar en forma agil algun documento en su enorme archivo y a la vez, éstos se deterioran inexorablemente

Mediante este nuevo sistema cada texto es recorrido por un lector óptico y a cada escala de blanco y negro y gris le es asignado un codigo digital que es almacenado con un rayo lásar en un video disco La lectura del disco, es simplemente el sistema inverso

El ahorro en espacio de almacenamiento, comparado con la microfilmación es uno de los elementos que empujaron a la biblioteca a optar por los videodiscos

El tiempo de acceso tambien disminuye en forma considerable, aportando además la capacidad de que varios usuarios tengan acceso al mismo documento simultaneamente pues cada terminal de lectura cuenta con su propio dispositivo de almacenamiento magnetico en el cual es archivado el documento que solicitó para ser revisado

Antes de grabar la informacion en los videodiscos ésta pasa por un terminal que va controlando la calidad de la reproducción Luego computadores IBM y Data General son utilizados para indexar los documentos codificar los videodiscos para su posterior recuperación

Comienza remezón en la industria da los semiconductorea

Intel Corp , el exitoso fabricante da semiconductores, conocido especialmente por sus microprocesadores 8088 8086 y 80286 utilizados ampliamente en los equipos IBM PC y compatibles anunció que dabido a una debil demanda deberá cerrar una planta de producción en Santa Cruz, California y despedir a 900 empleados un 4% del total

Esta medida de Intel, no hace más que seguir los pasos de otras importantes industrias del área tales como National Semiconductor que debio paralizer la producción por dos semanas en enero, Texas Instruments que anunció el despido de 2 000 personas y Honeywell que anuncio el despido de 1 000 más

A la raiz de estas medidas, se encuentra el fenomenal crecimiento en los niveles de producción de integrados en Japón, sumado a un notorio decanso en la demanda por parte de los fabricantes de computadores

Sin embargo las perspectivas son menos sombrias ya que a pesar de esa baja en la demanda paulatinamente ha comenzado a crecer el interés por la fabricación de integrados a la medida, es decir de acuerdo a los requerimientos especificos del cliente

Si bien por ahora esta actividad no representa un porcentaje significativo de las ventas de los principales fabricantes de semiconductores de acuerdo a National Semiconductor estas representarán alrededor de un 30% del total en 1990

Sierra: La nueva generacion da IBM

IBM dio a conocer recientemente, los primeros modelos de su nueva serie de gigantescos computadores, conocida previamente por el nombre de Sierra y ahora oficialmente Series 3090

El primer equipo de esta serie en ser comercializado, sera el modelo 200, que se espera salga a la venta en noviembre próximo Este, en capacidad (alrededor de 30 MIPS) es similar a otros equipos que actualmente comercializa IBM El segundo modelo, el 400, que casi duplica en capacidad al 200, recién saldrá a la vente a mediados da 1987

Supermini da Prime

Prime Computers anunció un nuevo modelo de supermini, el 9955 que es un cincuenta por ciento mas podaroso qua su predecesor y costará alrededor de un 7% menos

El 9950 es capaz de procesar cuatro millones de instrucciones por segundo y el precio básico as da US$ 321 000

Bancos inglases organizan club.

Diez instituciones financieras británicas formaron recientemente un club computacional cuyo objetivo es investigar y desarrollar sistemas capaces de estimar riesgos y entregar consejos respecto a posibles inversiones

El primer sistema al que se ha abocado el club, servirá para determinar la situación de compañías en diferentes sectores, como ayuda para la evaluación posterior de créditos

El presupuesto inicial del club es de 250 000 libras de las cuales el proyecto Alvey (organismo gubernamental dedicado al desarrollo de tecnologia informática de quinta generación) aportará la mitad

Los objetivos finales del club son incentivar dentro de la comunidad financiera el uso de sistemas expertos, familiarizar los con estos y desarrollar pequeños sistemas piloto que sirvan como base a desarrollos futuros

USUARIOS CROMEMCO

Dada la escasa información existente respecto a los computadores Cromemco ha surgido una iniciativa de parte de Dario Carrasco M de formar un pool de direcciones de todos aquellos usuarios de Cromemco que esten interesados en compartir la información de que dispongan

Si usted dispone de material que vaya mas alla de un manual Basic sean compiladores perifericos sistema operativo etc o desea comunicarse con otros usuarios para recabar mayor información llame a Dario Carrasco al fono 763248

PC-Jr: A peaar da todo se discontinúa.

Ante la sorpresa de todo el mundo, IBM anunció finalmente que terminaba con la producción del PC Junior, y que si bien continuará apoyando el producto con software, respuestos y todo lo necesario para los usuarios, no tiene prevista ninguna fecha para reiniciar su producción

Naturalmente, la noticia resultó sorprendente en al medio, pues en la pascua solamente se vendieron casi 200 000 de estos equipos, luego de las mejoras que hizo IBM en su teclado y memoria Al mismo tiempo IBM rebajó su precio, llegando éste a menos de US$ 800 durante la pascua Sin embargo, al terminar IBM con su politica navideña de descuentos, las ventas del PC Junior bajaron abruptamente



Postergado el lanzamiento de Jazz para el Macintosh

En junio, recién será liberado Jazz, la versión para el Macintosh del exitoso software Lotus 1,2,3

Si bien en enero ya fueron presentadas al público versiones finales del producto, aun faltaban algunos detalles que solucionar, los que se esperaba estuviesen resueltos en abril, pero, lamentablemente, para Apple esto no fue así

En efecto, para Apple es de extrema urgencia que ese paquete de software sea liberado, pues le permitirá al Macintosh entrar con propiedad a competir con IBM en los terrenos en que el IBM-PC es fuerte, sumando a su favor las cualidades propias del Macintosh

De acuerdo a cifras publicadas por la empresa consultora norteamericana Future Computing si bien Apple ha logrado introducir sus equipos a las grandes corporaciones en alguna medida, ésta es aun muy pequeña

	PC y PC-XT (1 800 000 uds)	Macintosh (292 000 uds)
Escuelas	2%	17%
Oficinas medianas	18%	5%
Oficinas pequeñas	20%	13%
Hogares	20%	49%
Grandes Oficinas	40%	16%

De la tabla se puede apreciar que en las grandes cuentas, en unidades Apple ha logrado vender alrededor de un 6% de las unidades que ha vendido IBM Precisamente para revertir estos indices. Apple ha apoyado la aparición de productos tales como Jazz, su nueva impresora Laser y su red (Mac Office) que permite interconectar hasta 32 computadores



Nueva guerra de precios en perspectiva

Como es ya tradicional a estas alturas del año, IBM se prepara a rebajar los precios del PC y XT, o así al menos lo aseguran observadores del mercado norteamericano

De acuerdo a ellos IBM ha rebajado usualmente los precios de sus equipos en esta época, para mantener el volumen de ventas en un periodo generalmente flojo, aprovechando así de mantener a la defensiva a sus competidores

Otra razón que podría llevar a IBM a reducir sus precios es la próxima aparición de un nuevo computador de su familia PC, el que es conocido por ahora bajo el nombre de PC2 junto a la inminente defunción del PC Junior (ver noticia aparte)

De acuerdo a los observadores, el PC2 sería una versión mejorada del PC con un procesador similar al 80186 del PC-AT con capacidad de multitarea IBM se vería obligado entonces a bajar el precio del PC para hacer lugar al PC2 entre éste y el AT

Sin embargo no sólo IBM estaria preparándose a una rebaja de precios, sino que es el ambiente generalizado en el mercado norteamericano debido a los abultados inventarios con que han quedado las principales empresas dada la baja en las ventas Apple debió cerrar cuatro plantas de producción durante una semana y del mismo mal se han quejado Digital, Wang y Data General

Sperry abandona sus planes en Brasil

En un numero anterior, dábamos cuenta del traspaso de Sperry en Brasil a manos de una compañía local, como forma de superar las trabas a los productos importados impuestas por la Secretaria Especial de Informática

El plan original de Sperry, consistia en vender sus instalaciones en Brasil a Novadata Sistemas e Computadores, para luego hacerla a ésta representante de la línea Univac y fabricar computadores (minis y micros) compatibles con Univac

La decisión de abandonar estos planes se debió sobre todo a los airados reclamos de los clientes de Sperry, quienes no confían que una pequeña empresa, formada sólo en 1979, posea el capital tecnológico y financiero necesario como para prestar los servicios de mantención necesarios para sus equipos

La computación imprime rapidez a la contratación da teléfonoa

De un mes a una semana –en algunos casos a una hora– disminuirá el tiempo que media entre la consulta del interesado en adquirir teléfono y la respuesta sobre la eventual contratación El moderno sistema computacional que permite este notable acorte de la espera entró en servicio en abril recién pasado en las oficinas comerciales que la Compañia de Teléfonos de Chile posee en El Llano, Las Rejas y Ñuñoa, en la Región Metropolitana, y Arica en la Primera Región

El moderno sisteme, bautizado con el nombre de Sistema Integral de Atención al Publico (SIAP), se aplicó en forma piloto desde fines de 1983 en la oficina comercial "Apoquindo" Paulatinamente se irá extendiendo a todos los otros locales que la Compañia posee en zonas

Los terminales de computación del SIAP están conectados a un computador central de la Compañia ubicado en San Martín 50 De este modo es posible, por ejemplo, cumplir con las asignaciones de numero y de planta externa o línea, los dos requisitos técnicos básicos para optener teletono en forma rápida

En algunos casos se puede atender en una sola visita al futuro abonado puesto que, además el sistema proporciona las opciones para elegir el tipo de equipo que interesa al potencial suscriptor y calcula de inmediato los valores y el pago de cuotas si es necesario Luego se emite la orden de instalación que va al departamento respectivo y se imprime el contrato

El SIAP entrega, igualmente, rápidas respuestas a las consultas de las personas ya inscritas que van a preguntar sobre la techa de instalación del servicio y a las reclamaciones de los abonados También permite el conocimiento instantáneo de la situación legal que afecta a cada teléfono

Segun el gerente general de la Compañia de Teléfonos de Chile, Ivan Van de Wyngard, este sisteme pone a Chile en la vanguardia en cuanto a aplicar modernas técnicas computacionales para atender consultas de los usuarios

Se base el SIAP en dos computadores interconectados el IBM 3031 para su operación en linea y el IBM 4381 (uno de los más grandes de Chile) para las funciones batch

La capacidad de memoria del primero es de cuatro mega y la del 4381 de 16 mega Los discos con los que operan los computadores tienen una capacidad de almacenamiento de 8 360 mega

La red entre los terminales y el computador es controlada por un IBM 3705, al cual se conectan 14 lineas para la misma cantidad de ubicaciones remotas En esas posiciones hay alrededor de 100 terminales


En su oficina comercial Apoquindo la Cía de Teléfonos ensayó durante un año el sistema SIAP

Computación Educativa

Un interesante trabajo de aplicación de la computación a la educación está realizando un equipo multidisciplinario en el Taller Interfaz

Este grupo, en el que se cuentan educadores sicólogos y especialistas en computación desarrolla actividades para niños entre 8 y 12 años El objetivo de estos talleres es introducir a los ninos al mundo de la cibernética y su aplicación en sistemas naturales y organizaciones humanas comprender los principios básicos de los sistemas computacionales e inculcar una visión desmistificada y humanista de la potencialidad de la computación

Lo interesante de esta experiencia es que al mismo tiempo se busca desarrollar las habilidades de lectura sintesis e interacción grupal mediante tecnicas de dinamica de grupos Los talleres tienen una duración de 36 horas

Para mayor informacion dirigirse a Los Claveles 2745 Providencia o al fono 2318727

Ventas Digital

Entre las ultimas ventas de equipos Digital, realizadas por la firma Sonda, figuran un VAX-11/730 adquirido por la fábrica de ropa El As y un VAX-11/750 comprado por la empresa Lever Chile

El primer modelo está dotado de un MB de memoria y 131 MB en disco Además, incluye dos terminales de video y una impresora

Por su parte, el VAX-11/750 comprado por Lever Chile, más poderoso, dispone de una memoria de 3 MB y puede manejar un disco 1.4 GB con respaldo en cinta magnética y una impresora de 600 lineas por minuto La adquisición se completó con 14 terminales de video, seis terminales gráficos tres impresoras, dos plotters y 10 microcomputadores Rainbow 100 +

NCR
Innovadora tecnología computacional



ERA DE ESPERAR QUE EL COMPUTADOR PERSONAL NCR FUERA ASI.

Con todas las ventajas de la tecnología y experiencia de NCR.

- Procesador de datos de 8 bits o de 8/16 bits
- Procesador destinado a gráficos
- Pantalla con alto poder de resolución
- Teclado fácil de utilizar (numérico y alfanumérico)
- Memoria principal de hasta 512 KB.

Fácil de manejar y con un alto rendimiento en el procesamiento de datos.
Era de esperar de NCR, una empresa que cumple 100 años innovando.
Lo esperamos con nuestra red de distribuidores

NCR
Innovadora tecnología computacional

Mac-Iver 370 - Fono 380013 - Santiago

Donación a Liceo

Al comenzar el año escolar la NCR donó un minicomputador al Liceo Técnico Industrial A 107 de La Granja. En la oportunidad estuvo presente el Subsecretario de Educación, René Salamé.

Además de su conformación básica el equipo obsequiado cuenta con una impresora de 125 líneas por minuto. Tiene 64 KB de memoria y 10 MB en disco.

Este microcomputador será usado para apoyar la gestión docente y control curricular. En una segunda etapa se aplicará a la enseñanza de la tecnología computacional.

La Escuela de Informática y Negocios de la NCR puso becas a disposición del Liceo nombrado para que dos docentes estudien computación y operación de computadores en sus Aulas.



Asistentes a la entrega de un computador NCR al Liceo A 107 de La Granja. De izquierda a derecha Guillermo Simonet, gerente de Marketing de la NCR; Juan Rivera, de la Sociedad de Fomento Fabril; René Salamé, Subsecretario de Educación; Julio Lecourant, gerente distrito de industria de la NCR y Nicolás Alumohoi y Federico Montes Lira, de la Sofofa.

Batalla del escritorio.

Como un desafío a la competencia fue calificado por Sisteco el lanzamiento de su nueva arma en la batalla por la conquista de los escritorios.

En efecto, el Asistente Secretarial Wang (Wang Office Assistant) es un producto diseñado específicamente para la automatización de oficinas, integrando las funciones de un procesador de texto dedicado, una versátil máquina de escribir junto al poder de un computador personal.

Basado en un procesador Intel 60186, el Asistente cuenta con 256K RAM, un drive de 5 25 doble faz, con capacidad para 360K, dos puertas RS 232C y una puerta para un segundo drive opcional.

Como procesador de texto, el Asistente viene con WP Plus y puede correr también programas tales como Multiplan, PFS Report y File, GCI GraphWriter.



Nuevo gerente en Burroughs

Elías de la Cruz fue nombrado gerente comercial de Burroughs en Chile. El cargo es de reciente creación.

Entre los propósitos del nuevo gerente está el de continuar consolidando la estructura de líneas de negocios que caracteriza la organización comercial de la empresa.

El señor De la Cruz es "bachelor en marketing" en Sydney, Australia. En Burroughs ocupó antes los puestos de gerente de automatización de oficinas y gerente sucursal de gobierno.

El sistema de líneas de negocios ha dado muy buenos resultados a la firma. Esto es especialmente notable en la línea de salud, donde se está traduciendo en numerosas nuevas contrataciones.

Lógica introduce supermicro

En abril, Lógica dio a conocer el nuevo super-micro que se agrega a la línea de computadores MAI Basic Four.

Basado en un procesador Motorola 68000, el MAI 2000 viene con 1 Mb de memoria principal, disco fijo de 22 mega, expandibles a 120, respaldo en cinta de 43 mega y opcional un drive de floppies de 655 Kb. En resumen, un mainframe de hace sólo algunos años.

Naturalmente, este es un equipo multitarea y multiusuario, acepta hasta 14 terminales y trabaja sobre un sistema operativo tipo Unix, el Boss/IX. Junto al equipo, fue liberada una red que permite conectar hasta 63 computadores con 10 terminales c/u, un sistema de cuarta generación, generador de aplicaciones y un transportador de aplicaciones desde los equipos Basic Four anteriores.

Breves de ECOM

ECOMCALC

ECOM ha desarrollado una planilla de Cálculo (ECOMCALC) orientada fundamentalmente a computadores de la línea IBM. Dicho paquete funciona bajo ambiente CMS y está disponible para los usuarios de ECOM y para quienes quieran llevarlo a sus instalaciones

Campeonato de Ajedrez

El Sindicato ECOM y la Rama de Ajedrez, en conjunto con la Revista MICROBYTE están interesados en realizar un torneo deportivo con todos los miembros del área de la Computación e Informatica

Mayores antecedentes pueden obtenerse en Rama de Ajedrez - ECOM Fonos 741670 - 740076

ECOM inicia cursos en CORFO

El Sindicato ECOM dictó un Curso de Introducción a la Computación y Lenguaje BASIC" destinado a 40 funcionarios de CORFO

Inauguró el Curso el Presidente de la Asociación de Empleados CORFO, Sr Nelson Párez,y el Presidente del Sindicato ECOM, Sr Pedro Sánchez, actuando como relator el Sr Juan Enrique Sierra

Seminarios en Ecom

Recientemente fue dada a conocer una extensa lista de cursos y seminarios que desarrollará Ecom, como parte de su rol en la capacitación durante este año

La lista incluye cursos dirigidos a usuanos, especialistas, encargados de personal y publico interesado en general

Entre los 50 cursos y seminarios que serán desarrollados, caben destacar seminanos de teleprocesamiento, técnicas de programación y análisis auditoría computacional, prevención de riesgos, microfilmación, diseño lógico, sistemas digitales, etc

Para mayor información respecto a estos cursos y seminarios, se puede llamar al fono 696 7663

Novedades H.P. en Olympia

Están en el país las tres ultimas novedades de H P en materia de disqueteras e impresoras Son ellas la nueva versión de la disquetera de 3 1/2 pulgadas, designada como H P 9122 D, y las impresoras modelos NP 165 y ES 80 RO

La versión renovada de la disquetera duplica su capacidad de almacenamiento a 710 K, bajando así el costo en la relación pesos por byte almacenados En otras palabras, con una disquetera doble se pueden almacenar 1,4 MB de datos o programas Vale 1 600 dólares Los poseedores del modelo anterior —el 9121 D— pueden ampliarlo de acuerdo a las características del 9122 D

La NP 165 es una impresora de matriz de punta, compatible con todos los computadores que actualmente se comercializan en el país Viene con una velocidad de 180 caracteres por segundo escritura bidireccional, con calidad carta, y graficadora Entre otras características posee, además, un buffer de 2 KB e interfaz RS 232 C y Centronics Su costo es de 800 dolares

El segundo modelo de impresora, el ES 80/RO, es también compatible con todos los computadores que se venden en Chile Su principal campo de aplicación está en el procesamiento de textos ya que es de marganta Su precio aproximado es de 600 dolares

Ventas

Interesantes ventas realizó Burroughs en los ultimos dos meses Sodimac le compró 51 equipos B-25 para su proyecto de Procesamiento Distribuido que tiene por objeto automatizar las operaciones en las sucursales que posee esta firma En el desarrollo del plan, Burroughs también participó en asesoria

Por otra parte Intec de la Avenida Santa María, y la Dirección de Presupuesto del Ministerio de Hacienda le hicieron sendas adquisiciones, Intec compró un computador de la serie B-1000 con un MB de memoria, 130 MB en disco y 12 terminales y el organismo publico, un computador B-1965 y 10 terminales En el pnmer caso el pnncipal atributo del equipo es el generador de sistemas Linc En el otro el elemento diferenciador lo constituye el procesador de palabras Linc

Una red de computadores instaló la firma Olympia en el Servicio Nacional de Capacitación y Empleo (SENCE)

La red consiste en cuatro microcomputadores Hewlett-Packard 150 B y 30 MB de almacenamiento y una impresora de 200 caracteres por segundo que será compartida por los cuatro usuarios El sistema interconectado es del tipo Ethernet desarrollado por 3COM Para esta red se han desplegado localmente los sistemas de información que permitan llevar un acucioso control de cursos, organismos de educación, empresas y participantes de todos los cursos aprobados por Sence Además se empleará en la gestión administrativa del organismo

Adelco instala red nacional

La firma Adelco adquirio a Texas Chile 11 minicomputadores Texas Instruments, modelo BS-300

Adelco, que se dedica a abastecer al comercio nacional de diversas mercaderías, armó con ellos una red nacional entre todas sus sucursales, conectadas a los dos computadores centrales Texas Instruments que ya tiene en Santiago

Los BS-300 pueden manejar varios terminales en forma simultánea En este caso están dirigidos a controlar toda la gestión comercial de la firma que los compró

SIEDUC

Sisteco y SAPLAE (Sistemas de Apoyo para la Administración de Empresas) liberaron recientemente en conjunto un sistema computacional dirigido al área de la administración educacional

El sistema SIEDUC, requiere de un computador profesional Wang y el software permite facilitar el proceso de matricula, confección de fichas escolares, remuneraciones, compatibilización de horarios, confecciona tests, procesa la contabilidad de acuerdo a las normas vigentes para ese sector y lleva el inventario de los bienes muebles de una unidad educativa

El valor del paquete completo es de US$ 8 500 y ya cuenta con un primer comprador Lo adquirió el colegio Villa Maria de Las Condes

Sisteco se mueve.

Con bastante éxito ha comenzado Sisteco 1985 A su lista de clientes se han agregado la Superintendencia de Valores y Seguros, la que adquirió un Wang VS 45 con 600 Mega de almacenamiento externo, 8 pantallas y dos Professional Computer El objetivo de esta compra es agilizar la fiscalización de las compañías de seguros y su propia administración interna

Seguros Continental, que tampoco se queda atrás, aumento su configuración adquiriendo almacenamiento adicional en disco más pantallas e impresoras con objeto de estar preparada a la eventual puesta en marcha del seguro automotriz obligatorio

Además, se incorporaron CIENES, la Caja Bancaria de Pensiones, Comber Williams y Cía , Protelum, Prodesa Cía Chilena de Inversiones y otros difíciles de detallar

Instituto de Sistemas reúne a Expertos de Computación

Durante cinco días, entre el 8 y 12 de abril permanecieron congregados en el Hotel Galerías especialistas de la computación e informática, pertenecientes a 80 entidades estatales y privadas participantes en el segundo "Instituto de Sistemas, organizado por IBM Chile con el respaldo del Centro de Sistemas de Gaithersburg, Maryland, Estados Unidos uno de los centros más importantes de dicha compañía El gerente general de IBM-Chile, Hernán Carvallo destacó como objetivo principal del evento, en su discurso de inauguración, 'el permanente espíritu de utilizar medios efectivos para producir constante transferencia de tecnología al país en ciencias de la computación e informática" En la oportunidad, mencionó también que dicha corporación realizo una considerable inversión en investigación y desarrollo que supera los 11 mil millones de dólares en los ultimos cinco años Dijo además, que IBM-Chile ha apoyado a la economía nacional en la negociación de exportación de cartulina para tarjetas dando ingresos al país por 144 millones de dólares desde que opera el convenio entre la empresa Laja-Crown, hoy Papelera y la IBM World Trade Corporation Se refirió, por otra parte, a la construcción del nuevo edificio, lo que representa para la empresa una inversión superior a los 12 millones de dólares

Entre otros temas las conferencias abarcaron consultorías sobre sistemas de oficina procesamiento de textos computador personal transferencia electronica de documentos conexión entre diversos equipos pequeños y otros



En la fotografía de izq a der Hernando Morales Asesor Legal General José Mutis autoridad informat ca de gobierno Hernán Carvallo Gerente General de IBM y Alberto Canola Gerente de Comunicaciones de IBM

Reorganización y nuevos sistemas de Sonda.

Como una forma de optimizar su atencion al usuario fue calificada en Sonda la reestructuracion de su Departamento de Asesoría de Sistemas

Este Departamento a cargo de Pedro Aracena, creo tres unidades encargadas de la instalación, mantención y administración respectivamente De este modo mediante una mayor especialización se espera poder brindar una mejor atención a los clientes de esta empresa

Al mismo tiempo de dar a conocer esta reestructuracion fue lanzado un completo paquete de sistemas de gestión que abarcan las áreas fundamentales del proceso administrativo de toda empresa

El sistema de gestión Sonda (SGS) que puede ser integrado en toda la linea de computadores Digital esta compuesto por paquetes individuales (SGS Sueldos SBS-Inventario, SGS-Contabilidad etc) los que tienen la capacidad de interactuar entre ellos para una mayor economía de proceso



Resolviendo problemas
misioneros y caníbales

Carlos Contreras M

En nuestros estudios de matemáticas de la escuela se nos acostumbró a considerer como problemas la aplicación de fórmulas que permiten, por medio de un cálculo directo obtener un velor adecuado al problema planteado Incluso es muy raro que un problema de esos cursos se pueda resolver de varias maneras diferentes en forma igualmente correcta En la vida real nunca se nos presenta un problema con esas caracteristicas Casi siempre debemos seleccionar entre varias alternativas, primero un pequeno grupo que resuelva el problema de cualquier manera válida y luego, de entre éstas, la que lo hace mejor

La maestria en cualquier profesión consiste en saber tomar, para su examen, un numero suficientemente amplio de elternativas para seleccionar de entre ellas El experto sabe partir de un conjunto muy grande de posibilidades y desechar rápidamente la gran mayoria de ellas que no satisfacen los requisitos del problema El neófito en la materia elegirá la alternativa que tiene a mano o bien quedará perplejo ante un numero excesivo de alternativas

Desde su nacimiento los computadores fueron candidatos a realizar este proceso de selección entre un gran numero de alternativas debido a su enorme velocidad, que aumenta ano a año, y a sus capacidades lógicas elementales Se pensó que si las operaciones elementales de un computador se realizan entre mil y diez mil veces mas rápido que las operaciones elementales del cerebro humano (comunicaciones entre neuronas), entonces se podria tolerar un factor grande de estupidez en la máquina para resolver problemas de complejidad similar a los que resolvia el cerebro humano La posibilidad de hacerlo resulta especialmente atractiva pera los cientificos que se ven obligados a realizar operaciones monotonas y ándas que consumen buena parte de su valioso tiempo El computador podia seguir trabajando durante la noche, el fin de semana e incluso todas las vacaciones

El metodo para revisar todas las alternativas de un problema se conoce como Algoritmo de Busqueda Exhaustiva y tiene un amplio campo de aplicación en criptologia y control automático En la competencia con la inteligencia humana este metodo ha sido una de las mayores desilusiones de la Inteligencia Artificial El cerebro humano posee un método sintético de aproximación a los problemas complejos que deja fuera de competencia al más poderoso computador Un quimico experimentado puede seleccionar unas cuantas alternativas para la estructura de un compuesto complejo a part.. ue unas pocas cualidades globales mientras que un programa de computación demoraria siglos en examinar las propiedades de una fracción de las estructuras posibles Las ilusiones que se hicieron los especialistas fueron ilimitadas por ejemplo se desarrolló un programa que se proponia a si mismo y luego demostraba teoremas a partir de axiomas y teoremas ya demostrados La perspectiva era clara si el programa se proponia todos los teoremas posibles en cada momento e intentaba demostrarlos por medio de todas las combinaciones posibles de axiomas y teoremas, entonces en un tiempo suficiente este programa reharia toda nuestra geometria y luego seguiria adelante en un desarrollo independiente de nuestra inteligencia

A pesar del fracaso de estos programas el metodo de busqueda exhaustiva tiene a su haber algunos logros importantes El mas espectacular me parece que es su aplicacion en la demostración del Teorema de cuatro colores que afirma que para colorear un mapa de manera que dos paises con frontera comun no tengan el mismo color, basta contar con cuatro colores diferentes El teorema, propuesto en 1852, fue demostrado en 1976 por K Appel y W Haken con un uso masivo del computador En un proceso sinergico ellos examinaron una gran cantidad de configuraciones hasta desarrollar metodos de reducción que demostraban el teorema Examinar todas las alternativas de reduccion es imposible para un ser humano por lo que los arbitros que debian aprobar la publicacion de la demostracion del teorema mandaron hacer un programa independiente para comprobar la veracidad de todas las alternativas de la demostracion Con lapiz y papel no hubiesen terminado nunca

Supongamos que mantenemos en una tabla las ofertas de los supermercados de la ciudad y el costo en tiempo y movilización de trasladarnos entre cada uno de ellos Una busqueda exhaustiva entre todas las alternativas de viajes entre ellos nos podria indicar la manera mas economica de comprar una lista de productos Si un computador es alimentado con el estado de congestión de todas las calles de una ciudad podria indicarle al chofer de una ambulancia o una bomba de incendio, el camino más rápido para llegar a determinado lugar examinando todas las rutas posibles En una central telefónica se puede determinar la ruta más eficiente para el conjunto de llamadas vigentes examinando todas las alternativas posibles En criptologia –el desciframiento de mensajes– es inevitable el examen de todas las alternativas dada la intencion expresa de ocultar el código

Los ejemplos sugeridos requieren un estudio particular en cada caso, pues el tiempo de proceso varia rápidamente con el tamano del problema, por lo que fácilmente puede tomar un tiempo prohibitivo la solución completa, siendo necesario hacer simplificaciones También se puede restringir el problema usando otros métodos o el criterio

de una persona y luego usar la busqueda exhaustiva para seleccionar entre un número limitado de casos posibles

Para explicar el método usaremos el ejemplo de un problema sutil, que puede poner en apuros a cualquier persona, que además es simpático y que se puede resolver con un computador modesto y el lenguaje Basic. En aplicaciones de control automático, en que el computador debe tomar una decisión en tiempo real, debe usarse el lenguaje de máquina para revisar un número grande de alternativas para elegir la mejor. Cualquier humano se aburriría pronto de revisar exhaustivamente las alternativas y preferiría actuar por intuición, lo que puede ser peligroso o ineficiente

Misioneros y canibales

Tres misioneros y tres canibales deben atravesar un río utilizando un bote para dos personas. Los misioneros, muy desconfiados, piensan que los caníbales aprovecharán cualquier oportunidad en que estén en mayoría para comérselos. ¿Cómo deben organizar los viajes en el bote de manera que nunca quede en una ribera un número de caníbales mayor que el de misioneros?"

Cortando tres palitos de fósforos se pueden hacer pruebas en una mesa. Si Ud. conoce ya o resuelve el problema planteado, intente su solución con 7, 8 o 10 viajeros, para llegar a estimar este programa que revisa ordenadamente todas las soluciones posibles hasta alcanzar la solución o demostrar que ésta no existe

Este problema se puede representar, al igual que muchos otros, como un árbol de decisiones (Fig. 1). Los nodos son la situación antes de hacer un viaje, mientras que las ramas son los cinco viajes posibles en el bote hasta el próximo nodo

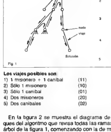


Fig 1

Los viajes posibles son

1) 1 misionero + 1 canibal	(11)
2) Sólo 1 misionero	(10)
3) Sólo 1 canibal	(01)
4) Dos misioneros	(20)
5) Dos canibales	(02)

En la figura 2 se muestra el diagrama de bloques del algoritmo que revisa todas las ramas del árbol de la figura 1, comenzando con la de más a la izquierda. Las flechas hacia abajo que hemos dibujado en el árbol equivalen a agregar un nuevo viaje al procedimiento examinado, ocurre cuando el programa sigue el bucle de la izquierda en el diagrama de bloques. Si el estado que resulta de ese viaje es ilícito, o inconveniente para los misioneros, entonces debe pasarse a probar con el próximo tipo de viaje, esto ocurre en el bucle de la derecha del diagrama. Si ya hemos probado con los cinco viajes, entonces quiere decir que el estado al que hemos llegado es un callejón sin salida y debemos deshacerlo (NODO = NODO − 1) y probar en este nodo el próximo tipo de viaje. Una razón importante para desechar un estado, o nodo, es que se repita exactamente un estado anterior en el procedimiento que estamos probando esto nos llevaría a una repetición indefinida de los mismos viajes. Finalmente si estamos en el primer nodo y ya hemos probado todos los viajes sin encontrar la solución, quiere decir que ésta no existe

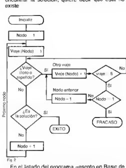


Fig 2

En el listado del programa –escrito en Basic de Sinclair– hay algunas diferencias por lo que se acompaña el diagrama de flujo de la figura 3 donde se indican las lineas correspondientes del listado. La cadena C$ de la linea 100 define los cinco tipos de viaje, el orden de éstos es importante para la eficiencia del programa. Ud. puede experimentar con un orden diferente, pero que no le falte ninguna de las alternativas pues son todas necesarias. La primera parte de la cadena, que se usa para dibujar a los viajeros sobre el bote, es más larga de lo necesario pues pensamos experimentar con un bote de mayor capacidad. En las lineas 130 y 140 se usa la función del Basic de Sinclair que asigna una parte de una cadena, para llenar las tablas T(I) y S(I) con el número de misioneros y de canibales en cada tipo de viaje, debe modificarse para otros Basics o, mejor, use la instrucción DATA para llenar las tablas T y S

El plan de viajes que se está probando en cada momento se almacena en la siguiente tabla
C(I), N° de canibales en la ribera izquierda
M(I), N° de misioneros en la ribera izquierda
B(I), ubicación del bote = 1 si está a la izquierda, = -1 si a la derecha
D(I), es el tipo de viaje que se hace
El índice I indica el nodo de que se trata

El número de alternativas para resolver el problema es igual a cinco elevado al número de viajes, cifra que crece muy rápidamente, para 6 viajes son 15 625 alternativas, para 9 viajes ya son casi dos millones. El problema desecha algunos viajes rápidamente, lo que se ve en la pantalla al ir cambiando la carga del bote, y se detiene al encontrar la primera solución. Luego de esto ofrece mostrar el procedimiento sin las vacilaciones del proceso de busqueda, o sea, sin volver a nodos anteriores como tuvo que hacer inicialmente

Fig 3

```
   5 FAST
  10 CLS
  20 PRINT AT 0,4;"misioneros y canibales"
  30 PRINT "       (CARLOS CONTRERAS 1983)"
 100 LET C$="CCCCCCCCCCMMMMMMMMMM11100120013"
 102 LET Z$=" "
 104 LET L$="(  >2^=E0"
 106 LET R$=Z$+"<=  )"
 110 DIM M(30)
 111 DIM C(30)
 112 DIM B(30)
 115 DIM D(30)
 116 DIM T(5)
 117 DIM S(5)
 120 FOR I=1 TO 5
 130 LET T(I)=VAL C$(10+2*I)
 140 LET S(I)=VAL C$(20+2*I)
 150 NEXT I
 155 SLOW
 160 PRINT "NUMERO DE VIAJEROS IR A 21:"
 170 INPUT N
 190 REM
 191 REM       << INICIALIZA >>
 192 REM
 200 LET CC=INT (N/2)
 210 LET MC=N-CC
 220 LET BP=1
 230 LET I=1
 240 LET CI=CC
 250 LET CD=0
 260 LET MI=MC
 270 LET MD=0
 300 GOSUB 2000
 310 LET C(I)=CI
 320 LET M(I)=MI
 330 LET B(I)=BP
 335 IF MI=0 AND CI=0 THEN GOTO 900
 337 REM
 338 REM        << OTRO NODO >>
 339 REM
 340 LET I=I+1
 342 REM
 343 REM      << PRIMER VIAJE  >>
 344 REM
 345 LET D(I)=1
 350 LET D=D(I)
 357 GOSUB 1000
 360 IF BP=-1 THEN GOTO 380
 370 IF CI<T(D) OR MI<S(D) THEN GOTO 520
 375 GOTO 390
 380 IF CD<T(D) OR MD<S(D) THEN GOTO 520
 385 REM
 386 REM      << HACE EL VIAJE >>
 387 REM
 390 LET CI=CI-BP*T(D)
 395 LET CD=CC-CI
 400 LET MI=MI-BP*S(D)
 403 LET MD=MC-MI
 405 LET BP=-BP
 410 IF MI>0 AND CI>MI THEN GOTO 500
 420 IF MD>0 AND CD>MD THEN GOTO 500
 422 REM
 423 REM    << SE HA REPETIDO ? >>
 424 REM
 425 FOR K=1 TO I-1
 430 IF CI=C(K) AND MI=M(K) AND BP=B(K) THEN GOTO 500
 460 NEXT K
 465 GOTO 300
 496 REM
 497 REM << DEVUELVE EL VIAJE >>
 498 REM
 500 LET BP=-BP
 503 LET CI=CI+BP*T(D(I))
 507 LET CD=CC-CI
 510 LET MI=MI+BP*S(D(I))
 515 LET MD=MC-MI
 516 REM
 517 REM  << PRUEBA OTRO VIAJE >>
 518 REM
 520 LET D(I)=D(I)+1
 530 IF D(I)>5 THEN GOTO 600
 540 GOTO 350
 596 REM
 597 REM << HAY QUE MODIFICAR NODO ANTERIOR >>
 598 REM
 600 PRINT AT 21,0;"VOLVEMOS ATRAS"
 610 LET I=I-1
 615 IF I<=1 THEN GOTO 800
 620 LET CI=C(I-1)
 623 LET CD=CC-CI
 625 LET MI=M(I-1)
 627 LET MD=MC-MI
 630 LET BP=B(I-1)
 633 GOSUB 2000
 650 GOTO 520
 800 SCROLL
 805 PRINT "   FRACASAMOS"
 810 STOP
 900 CLS
 903 PRINT "<<<E X I T O >>>",,"QUIERE VER COMO SE HACE ?(S/N)"
 905 INPUT Q$
 906 IF Q$="N" THEN STOP
 907 FOR N=1 TO I
 910 LET MI=M(N)
 915 LET CI=C(N)
 920 LET BP=B(N)
 925 LET MD=MC-MI
 930 LET CD=CC-CI
 940 GOSUB 2000
 945 LET D=D(N+1)
 947 GOSUB 1000
 950 NEXT N
 991 STOP
 996 REM
 997 REM << DIBUJA A LOS PASAJEROS >>
 998 REM
1000 PRINT AT 21,12-6*BP;"  "
1010 PRINT AT 21,12-6*BP;C$(1 TO T(D)) TO (10+S(D))
1020 RETURN
1996 REM
1997 REM << DIBUJA EL BOTE Y EL RIO >>
1998 REM
2000 LET B$=R$
2005 IF BP=1 THEN LET B$=L$
2008 SCROLL
2010 PRINT MI;"M ";CI;"C";"  ";B$;"  ";MD;"M";CD;"C"
2090 RETURN
```




DATAMERICA
Estado 139 - Fonos 722525-722572
Computador
Personal corona
• Full Compatible con IBM PC MR
• Alta resolución 640 × 400.
corona

CURSO

Uso del Sistema Operativo CP/M

J. Aravena L.

Temario

1. 1.1 Qué es un S.O. 1.2 Características de CP/M 2. Operaciones básicas 3. Nombres de archivos	1ª Parte
4. Comandos de CP/M Básicos Transitorios	2ª Parte
5. Detalle de algunos comandos transitorios: STAT, ASM, DDT.	

Estructura física de los archivos CP/M
6. Estructura del sistema operativo CP/M
7. Fabricación de programas usando CP/M

5. Detalle de algunos comandos

Uno de los aspectos más importantes de CP/M y que explica su popularidad entre los fanáticos de la computación, es que otorga facilidades para la creación de nuevos programas de sistema, es decir, de nuevos comandos que enriquecen la potencia de la máquina, la capacidad del mismo sistema operativo. En este sentido es un sistema muy "abierto", facilitando al usuario la creación del software.

Algunos Sistemas Operativos, generados por grandes empresas fabricantes de hardware, no otorgan este tipo de facilidades, quedando el usuario a merced de lo que el fabricante quiera venderle en cuanto a software básico, vale decir, existe un mercado cautivo para lenguajes y utilitarios de alto nivel. Al parecer se supone que el cliente cautivo es un buen negocio. Sin embargo, el usuario CP/M se encuentra con todas las herramientas necesarias para desarrollar su creatividad y así, rápidamente brotó todo tipo de software para CP/M, realizado por cientos de proveedores distintos.

Esto, probablemente, contribuyó más al beneficio de Digital Research, creadora de CP/M, que lo que el cautiverio de los clientes pudiese significar. Ojalá que saquen moraleja los creativos del mercadeo.

Entre las herramientas que entrega CP/M y que permiten el desarrollo de la creatividad del usuario, destacan el Ensamblador 8080 llamado ASM.COM y la Herramienta Dinámica para Depuración, llamada DDT.COM. La intención de este artículo es destacar algunos aspectos de estos utilitarios, de modo que principiantes puedan comprender su uso y seguir el manual de CP/M, que no es muy amistoso para los no-iniciados. En este número, también revisaremos el utilitario STAT.COM.

5.1. Comando ASM.

El ensamblador 8080 incluido en CP/M, permite realizar programas en lenguaje de máquina sin tener que lidiar con los confusos códigos hexadecimales, que limitan el tamaño del programa por nuestra incapacidad de llevar control de todas las alternativas de un programa complejo. No es la intención de este artículo traducir ni reemplazar la explicación dada por el manual CP/M para este utilitario, sino que comentar las características más relevantes para introducir al principiante. Tampoco es la intención incluir un curso de programación en lenguaje ensamblador. Para esto, remitimos al lector a los muchos textos y otros artículos de esta Revista que tratan estos temas.

Está fuera de lugar que ponderemos las ventajas del trabajo con un ensamblador, pero resumiremos las características más notables.

Desde luego, queda clara la comparación con la programación en lenguaje de máquina, en hexadecimal directamente, ya que ésta solo se puede utilizar en situaciones muy simples, en programas de poco tamaño. En efecto, el programar en lenguaje ensamblador permite liberarse de los códigos Hex que nada nos indican, utilizando siglas mnemotécnicas que nos recuerdan el significado de la instrucción. Así se escribe JMP para un salto del programa (Jum = saltar en inglés) en vez de un hexadecimal como C3 que nada evoca. Para nosotros, hispanófonos, tal vez lo lamentable sea que, precisamente las siglas sean inventadas en inglés, obligándonos a los no angiófonos a luchar con el idioma además de la computación.

Pero también el lenguaje ensamblador permite utilizar nombres o etiquetas simbólicas para referirse a los datos y a algunos parámetros, permitiendo que el programa sea más legible, comprensible para los humanos, lo cual tiene importancia, ¿no?

Uso del ensamblador

Para utilizar el ensamblador es necesario preparar el programa con ayuda de un Editor. El utilitario ED.COM puede servir, pero mejor es utilizar el WordStar u otro procesador de texto más moderno. (En este caso, el archivo debe ser preparado con la opción N, o sea, un no-documento). El nombre de este archivo debe, necesariamente, tener la extensión .ASM pues de esta forma es reconocido como un programa en lenguaje ensamblador.

En la formulación de un programa en ensamblador sólo se pueden emplear las instrucciones predefinidas y las pseudo-operaciones, como se indicará más adelante.

Si suponemos que el archivo que contiene el

programa se denomina EJEMPLO ASM, se podrá ensamblar, es decir, traducir a lenguaje de máquina mediante el comando

A) ASM EJEMPLO

si imaginamos que ambos, ensamblador y programa, residen en el disco A

La línea de código

El formato de un programa en lenguaje ensamblador es bastante rígido para el gusto de un programador BASIC pero debe recordarse que muchos años separan la creación de ambos lenguajes El formato básico es

etiqueta operación operando comentario

Los distintos campos de la línea deben estar separados por un espacio al menos

La etiqueta es optativa pero es indispensable si se realiza un salto a esa línea de instrucción El ensamblador reemplazará la etiqueta por la dirección correcta de la celda de memoria que corresponda

En caso de no existir la etiqueta la "operación" debe estar precedida de espacios en blanco para no ser confundida con una etiqueta

Los comentarios están precedidos de punto y coma y deben estar al final de la línea

Operandos

Los operando pueden ser símbolos definidos en otra línea del programa o bien pueden ser constantes numéricas directamente Los numeros pueden expresarse en forma decimal o si se explicita pueden usarse numeros binarios, octales o hexadecimales Para esto ultimo es indispensable que comiencen con un numero mayor que 10 y tengan un sufijo B O o H Un ejemplo aclarará lo ultimo

246	es un numero decimal
234H	es un numero Hex
34O	es un numero octal
100110B	es un numero binario

Están mal escritos y son rechazados, las siguientes constantes

278Q	(numeros octales no pueden usar el simbolo 8)
11021B	(numeros binarios sólo conocen el 0 y el 1)
FE00H	(numeros hex deben empezar con una cifra menor que 10 y por lo tanto debe escribirse 0FE00H)

Los operando de la instrucción pueden ser, a su vez, operados segun las aritméticas y lógicas usuales Es decir, un operando puede aparecer en una instrucción de la forma JMP ORIGEN + 12H, por ejemplo La constante ORIGEN debe estar definida en otra parte del programa

Si se requiere una constante alfabetica como operando, basta escribirla entre apóstrofes, así ESTO ES UN MENSAJE El ensamblador se preocupará de traducirlo a ASCII

En fin, existe un gran numero de facilidades para la programación que aquél que ha pasado por la programación en Hex, apreciará

Pseudoinstrucciones

Es importante aclarar que además de las instrucciones de máquina propia del ensamblador que son específicas de la CPU objeto del lenguaje, existen las Pseudoinstrucciones o directivos del ensamblador Estas tienen un formato similar a las operaciones pero contienen información para que funcione el traductor Una lista parcial de estas pseudoinstrucciones es

ORG xxxx	define el origen del programa en la dirección xxxx
END	especifica el fin del código
EQU valor	permite usar un símbolo para un numero Similar para efectos del ensamblado, al LET nombre = valor de BASIC El nombre se puede usar en todo el programa
DB expresiones	permite inicializar áreas de memorias con valores de un octeto Esto permite definir tablas de valores y mensajes en forma cómoda
DW expresiones	es similar a DB pero las expresiones son evaluadas como direcciones de 16 bits y almacenadas de la forma usual en 8080
DS expresión	sirve para reservar un área de memoria del tamaño dado por la expresión

Operaciones

Los códigos de operación aceptados por ASM-COM son los códigos estándar de la CPU Intel 8080 que el lector puede encontrar en un catálogo o en el mismo manual CP/M

Otras facilidades

El ensamblador ASM COM puede realizar algunos diagnósticos de error del programa en proceso Además puede generar un listado con extensión PRN que es más adecuado para tener una copia impresa con el código fuente y también en hexadecimal Existen otras facilidades más complejas, como por ejemplo el ensamblador condicional que no se explicará en esta oportunidad

A continuación se listan los errores que puede reconocer el ensamblador

D	no puede poner un elemento en un área de dato
E	expresión mal formada
L	etiqueta mal usada
N	facilidad no disponible en esta versión de ASM COM
O	expresión demasiado compleja
P	etiqueta que cambia de valor
R	operando incompatible con la operación
U	etiqueta inexistente
V	operando inadecuado

Uso práctico

En primer lugar se debe realizar un texto fuente, con código 8080, con ayuda de un procesador de texto El nombre del archivo debe tener la extensión ASM para poder procesarlo Supongamos que se ha elegido el nombre EJEMPLO ASM para ejemplificar

Luego debe ensamblarse mediante el comando

A) ASM EJEMPLO

Al terminar el proceso, se encuentran en el disco, además de EJEMPLO ASM que no se ha alterado, los archivos de salida EJEMPLO PRN para imprimir el resultado y examinar posibles errores, y el archivo EJEMPLO HEX.

Este último contiene el programa traducido al lenguaje de máquina, pero no es ejecutable directamente. En realidad se encuentra en una manera conocida como "Formato Hex de Intel", inventado en los años 70 cuando se trabajaba con teletipos.

El utilitario LOAD COM, que muchas veces se habrá preguntado para qué sirve, es el que permite convertir el archivo EJEMPLO HEX, en formato hex, en un archivo EJEMPLO COM que puede ser ejecutado. Para que esto pueda ser realizado bajo control de CP/M es indispensable que el programa tenga su origen en la dirección 100H. El ejemplo desarrollado en el manual CP/M, aclara varios problemas.

5.2. Utilitario DDT.COM.

Con el humor californiano habitual, este utilitario se denomina formalmente "Dynamic debugging tool", lo que se puede traducir como "Herramienta dinámica para la depuración". Pero su sigla DDT alude a un insecticida, jugando con la idea que los errores de los programas se denominan "Bugs", algo así como "bichos". De modo que DDT sirve para terminar con los bichos, naturalmente.

Herramientas incluidas

En realidad este utilitario trae varias herramientas incorporadas:

– Un Pequeño Ensamblador para realizar correcciones pequeñas sobre la marcha, sin recurrir a ASM COM.

– Un Desensamblador que permite examinar en códigos mnemonicos 8080 un párrafo de lenguaje de máquina, haciendo más comprensible su objeto.

– Un Monitor que permite examinar y alterar directamente contenidos de memorias y correr un programa.

– Un Simulador, y esto es lo mas poderoso, que permite ejecutar en forma supervisada un trozo de código de máquina. A medida que se realiza cada instrucción, se despliega en la pantalla el contenido de los registros y de los "Flags" o "Señas" de la CPU.

Para invocar este programa depurador se necesita escribir:

A) DDT 'archivo'

en que el 'archivo' a examinar debe ser tipo HEX o COM.

Lo que llama la atención, al invocar este comando, es que queda esperando órdenes, al igual que PIP o que el mismo CP/M. El símbolo que aparece indicando la espera de órdenes, es un guión (Este símbolo se denomina "incitación", "Prompt" en inglés). Para terminar la ejecución de este programa se debe teclear "Cntrl-C".

Las distintas órdenes que se pueden dar son las siguientes:

–Ensamblador	
Axxxx	comienza a ensamblar desde la dirección xxxx en hex
–Desensamblador	
Lxxxx	desensambla desde la dirección *xxxx* en *hex*
–Monitor	
Dxxxx	despliega a partir desde xxxx en hex y en ASCII
Fxxxx yyyy cc	llena la memoria desde xxxx hasta yyyy con la constante cc
Mxxxx yyyy dddd	mueve la zona desde xxxx a yyyy hasta la zona que comienza en dddd. Ojo con los traslapos
Sxxxx	substituye desde la dirección xxxx lo que se ponga desde el teclado. Para terminar se debe entrar un punto
G	Comenzar a ejecutar el programa sin supervisión
–Simulador	
Tn	Trazar la ejecución del programa durante n instrucciones a partir del valor actual del registro P – Contador de Programa. Trazar significa que es muestra paso a paso el contenido de todos los registros y flags
Xr	Mostrar y alterar un registro r. El símbolo r puede ser: P – Contador de programa S – Puntero del Stack (Pila) H – El par HL D – El par DE B – El par BC A – Acumulador También se pueden cambiar las señas: C – Acarreo Z – Cero M – Menos E – Paridad par I – Acarreo intermedio
Un	No trazar la operación de la CPU durante n instrucciones. Es útil cuando hay grandes zonas de programa que no requieren ser examinadas, de modo de no perder tiempo y concentrarse en el área conflictiva

Comentario

Todas las operaciones aquí indicadas tienen variantes y complementos que hacen aun más poderosas estas herramientas. Solo se muestra el aspecto fundamental para que el lector se introduzca en el tema y desarrolle sus propios ejemplos. Como es sabido, solo con la experimentación directa se llega a aprender, de modo que esta guía pretende romper el hielo inicial y vencer la timidez del experimentador.

5.3. Utilitario STAT.COM.

Lo sorprendente de este utilitario reside en lo heterogeneo de las operaciones que realiza. Su nombre evoca la información estadística que puede entregar, pero salvo en los casos más simples permite acciones bastante complejas. Existen varias formas para utilizar este comando.

En su primera forma:

A) STAT	Que efectivamente proporciona una estadística de la ocupación de cada lectora de diskettes, indicando el número de Kilobytes ocupados y si está disponible para la escritura del diskette. Esto ya se explicó en un numero anterior. Pero existen otras formas de emplear este utilitario indicando una línea de comando es

	pecificando el tipo de tarea que deseamos que STAT realice Un ejemplo de este uso es la forma
A) STAT * *	Que nos entrega un listado de todos los archivos como DIR pero con indicación del tamaño de cada uno

Otra forma que es de utilidad relativa es

A) STAT A − R/O	Que permite dejar protegido el disco A de modo que no pueda ser escrito (R/O significa READ ONLY sólo lectura) Si posteriormente se intenta escribir ese diskette el sistema de un mensaje de advertencia pero provoca una recarga del sistema con lo que el dispositivo puede ser escrito en un segundo intento

Más utiles son las formas

A) STAT archivo $R/O	Que protege permanentemente este archivo de posibles alteraciones
A) STAT archivo $R/W	Que desprotege un archivo R O dejándolo apto para lectura y escritura (READ WRITE)
A) STAT archivo $SYS	Que define a este archivo como un archivo de Sistema y por lo tanto no aparecerá al pedir el DIR de la unidad
A) STAT archivo $DIR	Que restituye la visibilidad del archivo tipo SYS ya definido
A) STAT B DSK	Permite revisar los parámetros característicos que tiene CP M reservado para cada lectora de diskettes Este comando debe hacerse para conocer la estructura de los diskettes que se usen

Hay otras formas más especializadas, que no ahondaremos en este curso introductorio La más sorprendente es

A) STAT VAL	Que emite una lista de correspondencia entre los cuatro dispositivos lógicos que reconoce todo sistema CP M y su correspondencia con los 12 dispositivos físicos que pueden o no estar definidos para una configuración real particular Este listado también se puede alterar con este mismo comando haciendo por ejemplo que todo lo que va a la pantalla se desvíe a la impresora, o viceversa La expresión
A) STAT DEV	Muestra la correspondencia que en el momento de consulta esté implantada entre los dispositivos físicos y los lógicos Esto está determinado por el llamado IOBYTE que se verá mas adelante

En resumen STAT es una herramienta que permite conocer los parámetros estadísticos de la instalacion y además proteger archivos de la alteración o del examen en el directorio junto con controlar las rutinas de entrada-salida

Bibliografía

- Digital Research CP M Documentation
- CP M 2 2 Preface North-Star Computers
- Manual de Uso CP/M MPF-III
- CP M Primer
- The Soul of CP/M Computers and Electronic. Mayo-Agosto de 1983

Efecto especial

Julio Rojas G., V. del Mar

En primer lugar, les mando esta subrutina para el ZX-81 en código de máquina, cuyo objetivo es causar un efecto especial bastante interesante

Lo que en realidad hace es alterar el valor del registro del CPU, denominado "i", que es el que el hardware del ZX-81 utiliza para ubicar la dirección en memoria de los patrones de los caracteres que son posteriormente representados en la pantalla. En teoría, esta técnica se podría usar para generar caracteres definidos por el usuario, pero desgraciadamente sólo se interpretan los 5 bits menores de "I", siendo por tanto el numero máximo admitido 31 (El problema es que los patrones de caracteres deben estar en ROM, por la forma que opera la Interfaz Sinclair de video N del E)

El programa en sí es el que sigue

```
BASIC                    ASSEMBLER
 1 REM *****             LD A,0
10 POKE 16514,62         LD I,A
20 POKE 16515,0          RET
30 POKE 16516,237
40 POKE 16517,71
50 POKE 16518 201
60 FOR A = 0 TO 30
70 POKE 16515,A
50 RAND USR 16514
90 NEXT A
```

Tortugas

Aparte de eso, les envío una subrutina para el ZX-SPECTRUM, que permite efectuar los conocidos gráficos de tortuga

El programa lineas 5-40 dibuja una espiral de acuerdo a un ángulo dado

La subrutina permite hacer las siguientes instrucciones

LOGO

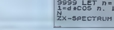
GO SUB 9990 = LIMPIA, debe siempre ser llamada al inicio de un programa.

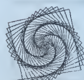
LET d = n GO SUB 9991 = AVANCE n

LET g =g + n= DERECHA n

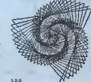
```
Listado

   1 REM *GRAFICOS DE TORTUGA**
   2 REM ***ESPIRAL GEOMETRICA**
***
   3 REM *******ENTRAR ANGULO***
***
   5 GO SUB 9990
   7 INPUT v
  10 FOR d=1 TO 130 STEP 1
  20 GO SUB 9992
  30 GO SUB 9991
  40 NEXT d
9989 STOP
9990 LET x=128: LET y=87: LET i=
0. LET g=0. RETURN
9991 GO SUB 9999. INVERSE i: PLO
T x,y: DRAW x1,y1: LET x=x+x1: L
ET y=y+y1: RETURN
9992 LET g=g+v: RETURN
9993 LET i=1: RETURN
9994 LET i=0. RETURN
9999 LET n=(450-g)*PI/180: LET x
1=d*COS n. LET y1=d*SIN n: RETUR
N
ZX-SPECTRUM
```

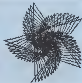




92



132



145

Frontenis

Mario Gutiérrez T.

El juego que les entrego (Frontenis) es muy similar al Squash, ya que en los dos el objetivo es rechazar una pelota que rebota en los lados y en el frente. Aun con la baja resolución gráfica del ZX81/TS1000 resulta éste un juego entretenido.

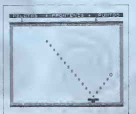
En los REMs he puesto el nombre de cada una de las subrutinas, es así, por ejemplo, como en la línea 200 se encuentra la subrutina "MOVIMIENTO DE LA PALETA" con su respectivo "mecanismo" de seguridad. Este evita que al llegar la paleta a los extremos izquierdo y derecho ésta cruce los valores, es decir, cambie la tecla de la derecha por la de la izquierda y viceversa.

Otra cosa interesante es el AUTORUN, ubicado en las líneas 630 y 635. Este sistema echa a correr el programa automáticamente apenas termina de cargarse.

IMPORTANTE: Para grabar un programa que contenga AUTORUN no debe usarse el sistema tradicional, sino que se debe tipear un GOTO a la línea que contiene al SAVE "...." (en este caso 630).

Mis agradecimientos al Sr. Mario Ramírez, quien me facilitó la impresora T/S 2040 para la ilustración y listado de esta programa.

```
  1 REM ******************
  2 REM *                *
  3 REM *   FRON-TENIS   *
  4 REM *                *
  5 REM * PARA MICROBYTE *
  6 REM *                *
  7 REM ******************
  8 REM
  9 REM
 10 PRINT AT 12,0; "FRONTENIS"
AT 21,0; "INGRESE N° DE PELOTAS
DE 1 A 10)"
 20 INPUT P
 25 IF P>10 THEN GOTO 10
 30 LET PEL=P+1
 35 LET PS=0
 40 LET PEL=PEL-1
 45 IF PEL=0 THEN GOTO 600
 50 CLS
 60 LET A=INT (RND*26)+1
 70 LET B=INT (RND*6)+4
 80 LET V=1
 90 LET W=1
100 LET Y=1?
105 LET  =20
110 PRINT AT 0,0; "PELOTAS * F
RONTENIS * PUNTOS"; AT 1,? PEL
AT 1,2? PS
112 PRINT
115 FOR I=  TO 20
116 PRINT AT I,0; " AT I,31 "
"
117 NEXT I
118 PRINT
120 GOSUB 200
130 GOSUB 500
140 GOSUB 300
150 IF B<>21 THEN GOTO 120
160 GOTO 40
197 REM
198 REM MOVIMIENTO PALETA
199 REM
200 LET X=X-(INKEY$="5")+(INKEY
$="8")
210 IF X<1 THEN LET X=1
220 IF X>28 THEN LET X=28
230 RETURN
297 REM
298 REM REBOTE EN LADOS
299 REM
300 PRINT AT B,A;
310 LET A=A+V
320 LET B=B+W
330 IF A=30 OR A=1 THEN LET V=-
V
340 IF B=3 THEN LET W=-W
350 IF B+1=Y THEN GOSUB 400
360 PRINT AT B,A; "O"
370 RETURN
397 REM
398 REM REBOTE
399 REM
400 LET P=A-X+1
410 IF P<1 OR P>3 THEN RETURN
415 LET PS=PS+2
416 PRINT AT 1,25;PS
417 PRINT AT Y-1,X; "
420 LET W=-W
425 PRINT AT Y-1,X;
430 RETURN
497 REM
498 REM PALETA
499 REM
500 PRINT AT  ,X-1; "
510 RETURN
597 REM
598 REM JUGAR DE NUEVO
599 REM
600 PRINT AT 10,7; "OTRO JUEGO ?
(S/N)"
610 IF INKEY$="S" THEN GOTO 30
620 IF INKEY$<>"N" THEN GOTO 60
0
625 STOP
630 SAVE "FRO"
635 GOTO 10
640 REM "FRONTENIS"
650 REM
660 REM M. GUTIERREZ
670 REM
680 REM   20 01 85
```



Simulación radiactiva

Alejandro Mardones

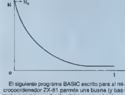
Quizás para muchos lectores el fenómeno radiactivo no es un suceso particularmente novedoso pues, con toda seguridad todos hemos oído o leído alguna vez algo acerca de este muy interesante tema.

La radiactividad natural es la clave para determinar las edades de las rocas, restos fósiles y arqueológicos, pinturas, etc. Por ejemplo, mediante la utilización de ciertos elementos como "relojes" radiactivos ha sido posible estimar la edad de nuestro planeta en unos 4.600 millones de años, y la del Sistema Solar en 5.000 millones.

Esencialmente, el fenómeno de la radiactividad, descubierto en 1896 por el físico francés Henri Becquerel, consiste en que muchos núcleos de átomos se desintegran espontáneamente emitiendo algunas partículas (por ejemplo, electrones). Por lo tanto, estos elementos radiactivos son inestables, y conforme se desintegran van formando nuevos átomos de un nuevo elemento. El número de desintegraciones por unidad de tiempo está regido por una ley estadística que predice que de una muestra de N_o átomos originalmente presentes de uno de estos elementos, al cabo de un cierto tiempo t el número N de átomos sobrevivientes viene dado por la fórmula

$$N = N_o e^{-kt}$$

donde e = 2,718..., base de los logaritmos naturales y k es una constante positiva llamada constante de decaimiento y es característica de la sustancia particular considerada. El gráfico de esta ecuación da la curva de decaimiento radiactivo del elemento y es de la forma que indica la figura siguiente



El siguiente programa BASIC escrito para el microcoordenador ZX-81 permite una buena (y bastante simple) simulación del proceso de decaimiento radiactivo haciendo uso del generador de números aleatorios.

```
10 REM SIMULACION RADIACTIVA
20 FOR N = 0 TO 21
30 PRINT "(32 espacios en video Inverso)"
40 NEXT N
50 POKE 16418, 0
60 PRINT AT 23,5; "SIMULACION RADIACTIVA"
70 PAUSE 500
80 UNPLOT RND * 63, RND * 43
90 GOTO 80
```

Para detenerlo debe oprimirse BREAK.

La línea 50 permite escribir en la línea 23 (normalmente sólo se puede hasta la línea 21).

Este programa simula la evolución de una muestra de material radiactivo para lo cual llena la pantalla con 2.816 puntos (64 x 44) para luego borrarlos espontáneamente al azar uno por uno desapareciendo el área oscura rápidamente al principio y después cada vez más lentamente hasta eventualmente desaparecer por completo luego de un tiempo suficientemente largo, lo que da un buen bosquejo del fenómeno de emisión radiactiva. En términos del formalismo matemático todo el proceso obedece en forma aproximada a una ecuación de la forma antes descrita, lo que significa que si se pudiera contar cada cierto tiempo t el número N de puntos negros que van quedando y luego se grafica N en función de t se obtendría aproximadamente la curva de decaimiento radiactivo.

En realidad lo que en la práctica se hace con muestras radiactivas es medir el número N de emisiones nucleares en intervalos regulares de tiempo t con un contador Geiger-Müller y luego graficar el logaritmo natural de estas cuentas, ln N, en función de t, obteniéndose una línea de puntos aproximadamente recta. Luego estos puntos son aproximados por la "mejor" recta que los representa calculada mediante lo que en Matemáticas se denomina Método de los Mínimos Cuadrados o Análisis de Regresión Lineal del cual ya apareció un artículo en el número 3 de Microbyte.

(Una versión en "negativo" del mismo programa es

```
10 PLOT 63 * RND, 43 * RND
20 GOTO 10
```

En ese caso, cada punto negro representa un átomo que explotó ... con un poco de imaginación).

N. del E.)



Colores

En la sección cartas de nuestro último numero, publicamos la solicitud de Fernando Rivas que deseaba saber cómo se hace para lograr más de 4 colores cuando se trabaja con un modo gráfico de alta resolución.

Nuestros ágiles sabuesos comenzaron de inmediato la investigación y ya podemos comenzar a adelantar algunos antecedentes respecto a una de las características más notables de los microcomputadores Atari, su manejo de colores.

En esta oportunidad, veremos algunas técnicas que permiten lograr un mayor numero de colores en modo gráfico 7, o así al menos creeremos cuando observemos la pantalla. Conviene aclarar que en realidad si bien el computador generará sólo cuatro colores, un uso apropiado de estos podrá generar otros muchos.

Mezcla de Colores

Cuando observamos algun bloque de la pantalla y la vemos de un color liso, si lo hacemos con mayor detención veremos que está formado por numerosos puntos dibujados de determinado color.

Un método muy utilizado para generar colores adicionales consiste en mezclar puntos de distintos colores, los que generarán un color diferente a los dos colores originales debido a un efecto visual que corresponde a la mezcla de colores.

Utilizando este método, en modo gráfico 7 que acepta sólo cuatro colores, podemos generar hasta 10 colores distintos: los colores primarios 1, 2, 3, 4 y las combinaciones 1-2, 1-3, 1-4, 2-3, 2-4 y 3-4.

Con un poco más de trabajo, incluso estos diez colores pueden ser aumentados varias veces. Piense por ejemplo en combinaciones que usan más de dos colores o crear une trama diferente (2 de rojo por uno de azul, por ejemplo). De este modo, en realidad es infinita la cantidad de colores posibles que podremos desplegar en un modo gráfico que originalmente aceptaba sólo cuatro colores.

En el programa 1 podemos ver una implementación de esta técnica, mezclando dos colores. Al correr el programa se divide la pantalla en dos partes, una con fondo blanco y otra con fondo negro y sobre ésas son dibujados los cuadrados.

Hacer la demostración sobre fondos de distinto color permite demostrar que visualmente si bien el color en ambos cuadrados es el mismo por efecto óptico se ven diferentes. Este es otro método para generar colores adicionales.

A continuación el programa pregunta por el color y luminosidad que queremos en cada uno de los cuadrados. El número de color está en un rango de 0 a 15 y el número de luminosidad entre 0 y 14, pero sólo números pares.

Luego de ingresados estos parámetros, los cuadrados superiores toman los colores originales y los inferiores se colorean con la mezcla, la cual es muy diferente a los colores originales.

El programa entra en un ciclo, permitiéndole así probar con todas las combinaciones que desee.

```
[illegible]
70 DRAWTO 20,[illegible]
[illegible]
[illegible] 2
[illegible] I=11 TO 15
[illegible] PLOT I,11:DRAWTO I,15
260 PLOT I+80,11:DRAWTO I+80,15
270 NEXT I
280 FOR I=21 TO 29 STEP 2
290 FOR J=31 TO 39 STEP 2
300 PLOT I,J:PLOT I+80,J
310 NEXT J:NEXT I
320 FOR I=22 TO 29 STEP 2
330 FOR J=32 TO 39 STEP 2
340 PLOT I,J:PLOT I+80,J
350 NEXT J:NEXT I
360 COLOR 3
370 FOR I=31 TO 35
380 PLOT I,11:DRAWTO I,15
390 PLOT I+80,11:DRAWTO I+80,15
400 NEXT I
410 FOR I=22 TO 29 STEP 2
420 FOR J=31 TO 39 STEP 2
430 PLOT I,J:PLOT I+80,J
440 NEXT J:NEXT I
450 FOR I=21 TO 29 STEP 2
460 FOR J=32 TO 39 STEP 2
470 PLOT I,J:PLOT I+80,J
480 NEXT J:NEXT I
490 PRINT "COLOR 1(0,15)BRILLO(PAR 0-14)
500 INPUT FC,FB
510 PRINT "COLOR 2(0,15)BRILLO(PAR 0-14
520 INPUT SC,SB
530 SETCOLOR 1,FC,FB
540 SETCOLOR 2,SC,SB
550 GOTO 490
```

Programa 1

Colores Alternados

Seguramente todos ustedes han jugado con un círculo de cartulina el cual se divide en una serie de trozos pintados de distintos colores y luego se hace girar a gran velocidad. Nuevamente se produce un efecto visual por el cual no vemos los distintos colores, sino que vemos un color diferente y uniforme.

Este mismo efecto lo podemos lograr con el computador, pero antes expliquemos un poco algunos fundamentos de cómo despliega colores en la pantalla el computador.

Si bien al mirar la pantalla vemos una imagen estable, en realidad el computador regenera la pantalla 60 veces por segundo, una velocidad a la cual no somos capaces de distinguir el movimiento.

Para esto, el Atari cuenta con dos microprocesadores. El 6502 que es el cerebro del sistema y un ayudante, el ANTIC que se dedica a controlar el manejo de pantalla. Para esto, el ANTIC cada sesentavo de segundo revisa una lista en la que aparece el color en que debe ser dibujado cada punto de la pantalla.

La técnica de alternación de colores se base precisamente en esto. Si hacemos que varíe el color en que es dibujado un punto cada sesentavo de segundo lograremos un efecto muy similar al que conseguíamos con el círculo de cartulina.

En el programa 2 vemos está técnica en acción. En este es una rutina en lenguaje de máquina la que permite alterar con la velocidad requerida la lista de despliegue.

Al correr el programa, al igual que en el caso anterior, éste pregunta por dos colores primarios con su respectivo brillo, los que son desplegados en dos rectángulos superiores mientras que el resultado de la alternación de colores es desplegada en un rectángulo inferior.

Naturalmente, utilizar estas técnicas en sus programas, requerirá de esfuerzos mayores de programación, pues es necesario llevar un control permanente de cuáles son los lugares en la pantalla en que deseamos los diferentes colores. Sin embargo, este mayor trabajo redundará en un mejor aprovechamiento de las capacidades gráficas de su computador.

```
[illegible]
[illegible]
[illegible] (PAR [illegible]
40 I[illegible]
50 PR[illegible] (PAR [illegible]
60 INPUT [illegible],SB
70 SETCOLOR 0,PC[illegible]
80 SETCOLOR 1,SC[illegible]
90 COLOR 1
100 FOR I=21 TO 39
110 PLOT I,21:DRAWTO [illegible]
120 NEXT I
130 COLOR 2
140 FOR I=51 TO 69
150 PLOT I,21:DRAWTO I,39
160 NEXT I
170 SETCOLOR [illegible],0,0
180 COLOR 3
190 FOR I=26 TO 54
200 PLOT I,51:DRAWTO I,69
210 NEXT I
220 POKE 36770,240
230 RESTORE
240 FOR I=0 TO 39
250 READ A:POKE 1536+I,A
260 NEXT I
270 DATA 7[illegible],128,72,169,0,141,10,212
280 DATA 141,24,208,169,20,141,0,2
290 DATA 104,170,104,64,72,138,72
300 DATA 169,0,141,10,21[illegible],141,24,[illegible]08
310 DATA 169,0,141,0,[illegible] 104,170,104,64
320 POKE 1540,PC*16+PB
330 POKE 1560,SC*16+SB
340 POKE 512,0:POKE 51[illegible],6
350 POKE 54[illegible]6,192
360 END
```

Programa 2

Desensamblador para Commodore 64

Eduardo Ahumada M.

Un desensamblador es un programa que permite traducir de Lenguaje de máquina a Lenguaje assembler, es decir, realiza la operación inversa a la de un compilador de Assembler. Este tipo de programas puede ser usado para dos cosas, la más interesante es examinar las rutinas del sistema operativo y/o las del Intérprete de BASIC y la otra es para analizar las subrutinas en lenguaje de máquina que a veces tienen algunos programas publicados en revistas de computación.

El programa que presento en el listado adjunto es una versión para el computador Commodore C-64, que fue adaptada de la versión original escrita para el VIC-20. Aquellos dueños de VICs que tengan problemas en adaptar el listado a sus máquinas, pueden solicitarme una copia por correo enviando un cassette o disketta a la dirección de MICROBYTE. En todo caso, es posible adaptarlo a cualquier computador que emplee la CPU 6502 o la CPU 6510 (que tiene las mismas instrucciones), para lo cual intentaré explicar cómo funciona el programa.

Antes de proseguir, deseo agradecer a todas las personas que me han escrito a través de MICROBYTE, y en particular a Roberto Andrade, quien proporcionó la idea inicial para llevar a cabo este proyecto.

Para entender mejor cómo funciona el programa, hagamos un pequeño repaso explicando el formato de las instrucciones en lenguaje de máquina.

Tabla I: Operandos de instrucciones de máquina 6502

Cod	Tipo de Operando	Bytes	Representación en Assembler
A	Inmediato	1	= nnn
B	Página Cero	1	nnn
C	Pág. Cero, Indexado por X	1	nnn, X
D	*Pág. Cero indexado por Y*	1	nnn, Y
E	Absoluto	2	nnn
F	Absoluto Indexado por X	2	nnn, X
G	Absoluto, Indexado por Y	2	nnn, Y
H	*Indirecto indexado por X*	*2*	*(nnn, X)*
I	Indirecto indexado por Y	2	(nnn) Y
J	Indirecto	2	(nnn)
K	Implícito	0	
L	Relativo	1	nnn

Las instrucciones en lenguaje de máquina 6502 están divididas en dos partes: un código de operación, y un operando. El código de operación ocupa el primer byte de la instrucción, y el operando puede ocupar dos, uno o ningún byte adicional, dependiendo del tipo de operando que se trate. En la tabla I hago un resumen de las características de los distintos operandos que pueda tener una instrucción en lenguaje de máquina 6502.

El código de *operación de la Instrucción indica* cuál es el tipo de operando que le sigue, es decir para la misma instrucción Assembler pueden haber varios códigos de operación dependiendo del *tipo de operando usado. Como el código de operación* ocupa sólo un byte, ello significa que el valor numérico del código va del 0 al 255.

El corazón del Desensamblador está formado por una lista alfanumérica, de 256 elementos, llamada I$. Cada elemento de esta lista corresponde a un Código de Operación y contiene la siguiente información:

"CXXX"

donde C es una letra que indica al tipo de operando que usa este Código de Operación, y XXX es el nombre de la instrucción Assembler correspondiente. Las letras que indican el tipo de operando están indicadas en la primera columna de la tabla I. Por ejemplo I$ *(0) contiene* un "KBRK", lo que indica que el Código de operación 0 corresponde a la instrucción BRK ("Break") del Assembler, usando operando implícito.

El algoritmo básico que emplea el programa es:

```
REPETIR
  (Pedir Dirección Inicial (P))
  REPETIR
    REPETIR 17 VECES
      (Imprimir P y código mnemotécnico)
      (Analizar e imprimir operando)
      (Incrementar P)
    CONTINUAR
      (Esperar que se presione una tecla)
  CONTINUAR si no se presionó F2
CONTINUAR si no se presionó F2
FIN
```

P es una variable que indica cuál es la próxima *dirección de memoria en donde se debe desensamblar.*

El programa que presento en el listado adjunto es una versión algo más sofisticada, y difiere del algoritmo descrito en que es capaz de retroceder para desensamblar la página anterior, o avanzar y desensamblar la página siguiente.

Para lograr el efecto de retroceso, es necesario que el programa recuerde cuál era la dirección de la primera instrucción desensamblada de cada página. Esta dirección es recordada en la tabla S,

que como tiene 10 posiciones, puede "recordar" hasta 10 páginas

El programa está formado por muchas subrutinas y un programa principal, veamos a continuación cuál es la función de cada subrutina

a) GOSUB 900
Posiciona el cursor en la fila F7, columna C7.

b) GOSUB 930
Espera que el operador presione una tecla, y devuelve el carácter correspondiente en T8$, y su valor ASCII en T8

c) GOSUB 950
Lee un valor numérico entero de largo máximo L7, a partir de la fila F7, columna C7, y lo retorna en X8$

d) GOSUB 5100
Pregunta a partir de cuál dirección se debe desensamblar, y devuelve este valor en DI

e) GOSUB 5200
Imprime el formato de pantalla para el listado Assembler

f) GOSUB 5000
Imprime encabezados y pies de página.

Las subrutinas siguientes analizan cada una un tipo distinto de operando, el programa principal determina a cuál debe llamar viendo el código correspondiente en la tabla I$. Estas subrutinas reciben como datos P apuntando a la posición de memoria que contiene el primer byte del operando, e Y, que contiene el valor del primer byte del operando. Al terminar de ejecutar estas subrutinas dejan a P apuntando al byte siguiente al operando, es decir al código de operación de la próxima instrucción.

g) GOSUB 300 Operando inmediato
h) GOSUB 310 Operando de página cero
i) GOSUB 320 Página cero indexado por X
j) GOSUB 330 Página cero indexado por Y
k) GOSUB 340 Operando absoluto
l) GOSUB 350 Absoluto indexado por X.
m) GOSUB 360 Absoluto indexado por Y
n) GOSUB 370 Operando indirecto indexado por X
o) GOSUB 380 Operando indirecto indexado por Y
p) GOSUB 390 Operando indirecto
q) GOSUB 400 Operando relativo
r) GOSUB 410 Operando implícito

Desde el punto de vista del operador, el desensamblador funciona en la siguiente forma

1) Al comenzar a ejecutar, el programa solicita la dirección de memoria a partir de la cual se desea comenzar a desensamblar. Para terminar la ejecución basta presionar la tecla F2

2) A continuación se comienzan a imprimir en la pantalla las instrucciones en Assembler, hasta completar una página (17 instrucciones). El formato de cada línea es:

Dirección Código de operación Operandos

Por ejemplo: 52001 LDA = 0

Por supuesto, existe la posibilidad de que alguna posición de memoria no contenga ninguna instrucción válida, en cuyo caso el programa la imprime usando el formato

Direccion CON "x" (nnn)

Donde CON es una abreviación de CONstante, nnn es el valor contenido en dicha posición y "x" es el caracter ASCII correspondiente.

Para ver la página siguiente se presiona F7, para retroceder a la página anterior se presiona F1. Para especificar otra dirección a partir de la cual se debe desensamblar, presionar F2, con lo cual volveremos a la pantalla descrita en (1)

```
10 DEFFNA(X)=PEEK(X+1)*256+Y
100 GOSUB2000 GOTO1000
300 PRINT" #"MID$(STR$(Y);2,3),
301 P=P+1 RETURN
310 IFX<>255THENPRINTY, GOTO301
312 J=1 IFI<35ORI>127THENI=46
313 PRINT" "CHR$(34),CHR$(I),CHR$(34)"    "J, GOTO301
320 PRINTY"■,X", GOTO301
330 PRINTY"■,Y", GOTO301
340 PRINTFNA(P),
341 P=P+2 RETURN
350 PRINTFNA(P)"■,X"; GOTO341
360 PRINTFNA(P)"■,Y", GOTO341
370 PRINT" ("MID$(STR$(Y),2,3)",X)", GOTO301
380 PRINT" ("MID$(STR$(Y),2,3)"),Y", GOTO301
390 PRINT" ("MID$(STR$(FNA(P)),2,5)")", GOTO341
400 X=Y IFX>=128THENX=X-256
401 PRINTP+1+X, GOTO301
410 RETURN
900 PRINT"■"LEFT$("■■■■■■■■■■■■■■■■■■■■■■■■",F7),SPC(C7); RETURN
930 GETT8$ IFT8$=""THEN930
931 T8=ASC(T8$) RETURN
950 GOSUB900 X8$="",PRINT"_■",
```

```
951 GOSUB930 IFT8=13ORT8=137THEN956
952 IF(T8<>157ANDT8<>20)ORLEN(X8$)=0THEN954
953 PRINT"█  █▄█", X8$=LEFT$(X8$,LEN(X8$)-1) GOTO951
954 IFT8>=48ANDT8<=57THENX8$=X8$+T8$ PRINTT8$"_█",
955 IFLEN(X8$)<L7THEN951
956 PRINT" ", RETURN
1000 GOSUB5100 C7=0 P=DI PS=0 ST(0)=DI GOSUB5200
1010 GOSUB5000 FORF7=2TO18 GOSUB900 PRINT"█"P, IFP>65535THENF7=19 GOTO1060
1015 X=PEEK(P) IFI$(X)=""THENI=X X=255
1020 PRINTRIGHT$(I$(X),3), P=P+1 Y=PEEK(P)
1030 ONASC(I$(X))-64GOSUB300,310,320,330,340,350,360,370,380,390,410,400 NEXT
1060 GOSUB930 IFT8=133ANDPS>0THENGOSUB1100 GOTO1010
1070 IFT8=136THENGOSUB1200 GOTO1010
1080 IFT8=137THEN1000
1090 GOTO1060
1100 PS=PS-1 P=ST(PS) RETURN
1200 IFPS=9THENDI=P PS=0 ST(0)=P RETURN
1210 PS=PS+1 ST(PS)=P RETURN
2000 DIMI$(255),ST(9) PRINT"J█" LB$="
2010 FORI=0TO255 READI$(I) NEXT RETURN
5000 PRINT"█----DESENSAMBLADOR----"  FORL=1TO19 PRINTLB$, NEXT RETURN
5010 PRINT"█"LB$,LB$,"                    █", RETURN
5100 GOSUB5000 GOSUB5010 F7=3 C7=0 GOSUB900 PRINT"TODAS LAS DIRECCIONES "
5110 PRINT"Y VALORES SON EN DECIMAL." F7=8 GOSUB900
5120 PRINT"DIR. INICIAL =>", F7=21 GOSUB900 PRINT"█F2 FIN█", F7=8 C7=16 L7=5
5130 GOSUB950 IFT8=137THENPRINT"J" END
5140 DI=VAL(X8$) IFDI>65535THENGOSUB900 PRINT"      "; GOTO5130
5150 RETURN
5200 GOSUB5000 GOSUB5010 F7=20 C7=0 GOSUB900 PRINT"█F1 PAG. ANTERIOR",
5210 F7=21 GOSUB900 PRINT"F7 PAG. SIGUIENTE", F7=22 GOSUB900
5220 PRINT"F2 FIN█", RETURN
6000 DATAKBRK,NORA,,,,BORA,BASL,,KPHP,AORA,KASL,,,EORA,EASL,,LBPL,IORA,,,,CORA,C
ASL
6010 DATA,KCLC,GORA,,,,FORA,FASL,,EJSR,HAND,,,BBIT,BAND,BROL,,KPLP,AAND,KROL,,EB
IT,EAND
6020 DATAEROL,,LBMI,IAND,,,,CAND,CROL,,KSEC,GAND,,,,FAND,FROL,,KRTI,NEOR,,,,BEOP
,BLSR
6030 DATA,KPHA,AEOR,KLSR,,EJMP,EEOR,ELSR,,LBVC,IEOR,,,,CEOR,CLSR,,KCLI,GEOR,,,,F
EOR
6040 DATAFLSR,,KRTS,NADC,,,,BADC,BROR,,KPLA,AADC,KROR,,JJMP,EADC,EROR,,LBVS,IADC
6050 DATA,,,CADC,CROR,,KSEI,GADC,,,,FADC,FROR,,HSTA,,,BSTY,BSTA,BSTX,,KDEY,,KTX
A,,ESTY
6060 DATAESTA,ESTX,,LBCC,ISTA,,,CSTY,CSTA,DSTX,,KTYA,GSTA,KTXS,,,FSTA,,,ALDY,HLD
A,ALDX
6070 DATA,BLDY,BLDA,BLDX,,KTAY,ALDA,KTAX,,ELDY,ELDA,ELDX,,LBCS,ILDA,,,CLDY,CLDA,
DLDX
6080 DATA,KCLV,GLDA,KTSX,,FLDY,FLDA,OLDX,,ACPY,HCMP,,,BCPY,BCMP,BDEC,,KINY,ACMP,
KDEX
6090 DATA,ECPY,ECMP,EDEC,,LBNE,ICMP,,,,CCMP,CDEC,,KCLD,GCMP,,,,FCMP,FDEC,,ACPX,N
SBC
7000 DATA,,BCPX,BSBC,BINC,,KINX,ASBC,KNOP,,ECPX,ESBC,EINC,,LBEQ,ISBC,,,,CSBC,CIN
C
7010 DATA,KSED,GSBC,,,,FSBC,FINC,BCON
```

DESENSAMBLADOR BASIC

Programación y control de proyectos

Segunda parte: El método PERT

Guillermo Beuchat
Ing. Civil Industrial U. de Chile

En el articulo anterior de esta serie, presentamos el método conocido como CPM (Critical Path Method), de amplia utilización en el control y planificación de obras. Además, se hizo una introducción teórica de las redes de actividades, se explicó la manera de expresar las relaciones de precedencia y se mostró la manera de calcular el camino critico de un proyecto. Este ultimo se definia como una secuencia de actividades tales que el atraso en la ejecución de cualquiera de ellas con respecto al tiempo planificado, producia un atraso en el tiempo total (critico) del proyecto.

En esta segunda parte, abordaremos un método de análisis de redes con tiempos no-deterministicos Este método, al igual que el sistema CPM permite obtener una "ruta crítica" para un proyecto determinado aunque ésta no corresponde conceptualmente a la misma ruta en el metodo CPM Segun la lógica del método PERT, no existe una ruta critica, en lugar de ello, cada actividad del proyecto tiene una determinada probabilidad de encontrarse sobre la ruta critica En efecto, la ruta critica es aleatoria cuando los tiempos de las actividades son inciertos, por lo que para efectos de cálculo, algunas actividades tendrán una probabilidad cercana a cero de encontrarse sobre la ruta critica, y otras tendrán una probabilidad cercana a uno

El método PERT, cuyas siglas provienen del inglés Program Evaluation and Review Technique, es un método de análisis de redes de proyectos que fue desarrollado en los años 50 para llevar a cabo el proyecto del submarino POLARIS de la Armada de los Estados Unidos de Norteamérica Durante este proyecto, fue necesario programar y coordinar a más de 3.000 contratistas, proveedores y un gran numero de personas Se estima que su utilización permitió adelantar la fecha de terminación del proyecto en casi dos años

En proyectos relacionados con la investigación y desarrollo tecnológico, es muy comun encontrar serios problemas para estimar la duración exacta de cada actividad dentro del proyecto Esta incertidumbre hace prácticamente imposible la utilización del método CPM, por lo que se hace necesario introducir conceptos probabilísticos

Estimación de tiempos

El método PERT requiere de tres estimaciones de tiempo para cada actividad un tiempo pesimista, un tiempo más probable, y un tiempo optimista Mediante estas tres estimaciones, se introduce el concepto de incertidumbre en la duración de cada actividad Una suposición muy importante del método es que los tiempos reales de cada actividad tienen una distribución de probabilidad BETA La figura 1 muestra este tipo de distribución, que corresponde a un tipo de curva normal, pero sesgada hacia la derecha Esto indica que el tiempo pesimista tiene más probabilidad de ocurrir que el tiempo optimista Por otra parte, la distribución BETA permite calcular la duración esperada de cada actividad, a partir de la fórmula

$$Te = \frac{To + 4Tm + Tp}{6}$$

Como se puede apreciar, esta fórmula pondera el tiempo más probable Tm cuatro veces más que los tiempos optimista y pesimista El tiempo esperado de una actividad, calculado mediante esta fórmula, tiene una importancia fundamental: puede ser usado como tiempo deterministico, para aplicar un método de analisis similar al usado en la tecnica CPM Podemos entonces calcular a partir de las tres estimaciones de tiempo una ruta critica y los tiempos de iniciación y finalización para cada actividad Ello a su vez permite el cálculo de una ruta o secuencia critica de actividades y un tiempo total esperado

Distribución BETA

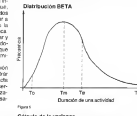


Figura 1

Cálculo de la varianza

Pese a lo anterior, no es tan fácil eliminar el problema de la incertidumbre en los tiempos de las

actividades. Si los tiempos de cada actividad son inciertos, el tiempo total de duración del proyecto también lo será. Para manejar este problema el método PERT supone que la varianza en el tiempo total de cualquier proyecto pueda calcularse sumando las varianzas de todas las actividades críticas de la red. Dado que la duración de cada actividad sigue una distribución BETA, podemos calcular su varianza a partir de la fórmula:

$$V = \left(\frac{Tp - To}{6} \right)^2$$

Esta fórmula se base en el supuesto de que los tiempos pesimista y optimista de cada actividad abarcen seis desviaciones standard de la distribución BETA, en que la varianza es el cuadrado de la desviación standard.

Cálculo del tiempo crítico

Una vez determinada la ruta crítica mediante el método CPM, a partir de los tiempos esperados calculados mediante la fórmula expuesta, es posible obtener el tiempo crítico total del proyecto. Si T es el tiempo total del proyecto, entonces su valor esperado E(T) corresponde a la suma de los tiempos esperados de todas las actividades sobre la ruta crítica:

$E(T) = \Sigma Te$

Por otra parte, la varianza del tiempo total corresponderá a la suma de las varianzas de todas las actividades sobre la ruta crítica:

$V(T) = \Sigma Vi$

Dados los valores anteriores, podemos decir que el proyecto tendrá una duración de

$E(T) \pm \sqrt{V(T)}$ [unidades de tiempo]

Otros parámetros del método

Al igual que el método CPM analizado en la primera parte de esta serie de dos artículos, el método PERT incluye los conceptos de holgura, iniciación tardía y finalización temprana para cada actividad. Teóricamente son lo mismo en ambos casos, pero en la práctica difieren. Por ejemplo, la holgura en una red PERT es una cifra útil sólo como indicación de su orden de magnitud, pues la incertidumbre del tiempo de cada actividad se reflejará también en una incertidumbre en el valor de la holgura. Por otra parte, los tiempos de iniciación tardía y finalización temprana también tienen valor relativo para la administración del proyecto, por las mismas razones.

Normalidad del tiempo crítico

Un supuesto muy útil que se puede hacer en el método PERT, es que la duración del proyecto, calculada según las fórmulas mencionadas, es una variable aleatoria con distribución normal o de Gauss. Por lo tanto, el método PERT nos permite calcular la probabilidad de terminar el proyecto en un tiempo dado, en lugar de entregarnos una fecha de terminación específica. La figura 2 muestra la curva normal, con sus parámetros expresados en términos del resultado de la red que nos interesa.

Distribución normal

Figura 2

En el programa BASIC adjunto se ha implementado un método para calcular el área bajo una curva normal, la que se ajusta a los valores correspondientes de media y desviación standard para calcular así la probabilidad de terminar el proyecto en un tiempo dado.

Análisis de costos

Por la naturaleza probabilística del método PERT, resulta muy difícil asignar costos a cada actividad, especialmente si estos costos no son fijos y se basan en el tiempo real utilizado en cada una. El método puede, sin embargo, entregar una estimación del costo de la ruta crítica, calculándolo en forma similar al método CPM. Aunque el programa adjunto no incluye un cálculo de costos, los lectores podrán sin duda introducir el tratamiento de los costos de la manera explicada en el programa CPM.

El problema de la restricción de recursos

Para concluir este breve análisis teórico de la programación de actividades en un proyecto, es necesario aclarar algunos aspectos importantes que lamentablemente impiden muchas veces el uso de técnicas de análisis de redes.

En primer lugar, los lectores habrán notado que ambos métodos suponen que cada actividad es realizada por una persona o "recurso" diferente. En efecto, el concepto de ruta crítica implica que algunas actividades deben realizarse simultáneamente, lo que a su vez implica que la misma persona no puede estar asignada simultáneamente a esas actividades. Dada la suposición anterior, se comprende por qué los métodos trabajan usando el concepto de horas-hombre de duración para cada actividad.

En segundo lugar, cabe preguntarse si la definición de "actividad crítica" es apropiada. Si un proyecto requiere de 1 000 horas-hombre, y sólo se dispone de una persona para realizarlo, el proyecto necesariamente se demorará un total de 1.000 horas, aunque la red de actividades probablemente indique una duración crítica menor. Lo correcto en este caso es sumar linealmente los tiempos asignados a cada actividad, y no construir una

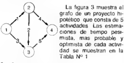
red. Por lo tanto, aparece el concepto de "recurso crítico o limitante" y las actividades críticas dejan de ser importantes.

Estas dificultades en la aplicación de los métodos de redes han llevado al desarrollo de complejos métodos generalizados, que ciertamente escapan al alcance de este artículo. Sin embargo, conviene conocer de su existencia y tener presente que hay paquetes de software disponibles para la planificación de proyectos complejos.

Utilización del Programa Basic

El programa adjunto, adaptado de una versión para el IBM-PC de L. POOLE, permite resolver una red de actividades mediante el método PERT, y presenta una estructura similar al programa CPM presentado anteriormente.

Grafo del proyecto



La figura 3 muestra el grafo de un proyecto hipotético que consta de 5 actividades. Las estimaciones de tiempo pesimista, mas probable y optimista de cada actividad se muestran en la Tabla Nº 1.

Figura 3

Datos de la red

Actividad	Nodo inicial	Nodo final	DURACION Optimista	Más probable	Pesimista
1	1	2	2	4	8
2	2	3	3	4	8
3	1	3	1	2	3
4	2	4	2	4	7
5	3	4	1	3	5

Tabla Nº 1

Para ingresar los datos de la red al programa, haremos uso de la conocida técnica de instrucciones DATA, en este caso a partir de la línea 2000 del programa. Como hemos mencionado anteriormente, este método permite obtener fácilmente un "archivo" para realizar cambios posteriores o un análisis de sensibilidad sobre la red sin necesidad de digitar nuevamente los datos. Para ello, basta grabar distintas versiones del programa con diferentes líneas DATA.

La figura 4 muestra el resultado obtenido con los datos del ejemplo. Como se puede apreciar, el tiempo crítico total se muestra incluyendo la desviación standard, en este caso

11 8 ± 1 5

El resultado muestra además dos tablas: una de actividades críticas y otra de actividades no críticas, con los datos correspondientes a cada una.

Actividadea críticas

Activ	Duración esperada	Desviación standard	Ultimo tiempo para empezar	Debe terminarse antes de
1	4 3	1	0	4 3
2	4 5	0 6	4 3	8 8
5	3	0 7	8 8	11 8

Actividadea no criticaa

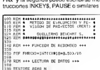
Activ	Duración esperada	Desviación standard	Comienzo temprano	Comienzo tardío	Final temprano	Final tardío	Holgura
3	2	0 3	0	6 8	2	8 8	6 8
4	4 2	0 8	4 3	7 7	8 5	11 8	3 3

Tiempo crítico 11 8 ± 1 5

Figura 4

Las líneas 870-1110 del programa permiten calcular la probabilidad de terminar el proyecto en un tiempo ingresado por el usuario, a partir de la curva normal de la figura 2. El área bajo la curva normal se calcula a partir de una conocida aproximación matemática.

El programa utiliza además una función de redondeo a una cifra decimal (línea 155), y dos líneas que detienen la ejecución hasta que el usuario presione alguna tecla para continuar (líneas 670 y 840). La primera función puede omitirse en caso que no se disponga de la instrucción DEF FNx y la segunda puede escribirse mediante instrucciones INKEY$, PAUSE o similares.



```
100 REM ********************************
105 REM *  METODO DE EVALUACION Y RE-  *
110 REM *  VISION DE PROYECTOS (PERT)   *
115 REM *                               *
120 REM *      GUILLERMO BEUCHAT S.     *
125 REM ********************************
130
140 REM +++ FUNCION DE REDONDEO +++
150
155 DEF FNR(X)=INT((X*10+0.5))/10
160
170 REM +++ LECTURA DE DATOS +++
180
190 RESTORE READ N
200 DIM A(N,2),S(N) F(N),E(N 2)
220 FOR K=1 TO N
240 READ A(K,1)   REM NODO INICIAL
250 READ A(K,2)   REM NODO FINAL
260 READ OP       REM TIEMPO OPTIMISTA
270 READ ES       REM TIEMPO MAS PROBABLE
280 READ PE       REM TIEMPO PESIMISTA
290 E(K,1)=((OP+ES*4+PE)/6)
300 E(K,2)=((PE-OP)/6)*((PE-OP)/6)
310 NEXT K
320
330 REM +++ CALCULO RUTA CRITICA +++
340
350 FOR K=1 TO N
360 IF S(A(K,2)) >= S(A(K,1))+E(K 1) THEN 380
370 S(A(K,2))=S(A(K,1))+E(K 1)
380 NEXT K
390 F(A(N 2))=S(A(N,2))
430 FOR K=N TO 1 STEP-1
440 IF F(A(K,1)) >= 0 THEN 470
450 IF F(A(K,1)) > F(A(K 2))-E(K 1) THEN 470
460 GOTO 480
470 F(A(K 1))=F(A(K,2))-E(K,1)
480 NEXT K
490
500 REM +++ RESULTADOS Y HOLGURA +++
505
510 PRINT CHR$(147)  REM --BORRA PANTALLA
520 V=0
530 C=0
540 L=0
550 PRINT TAB(10),"ACTIVIDADES CRITICAS"
560 PRINT TAB(10),"===================="
570 PRINT " ACT   DUR.ESP.  DES.STA,  UTPE    DTAD"
580 PRINT " ---------------------------------------"
590 FOR K=1 TO N
600 H=F(A(K,2))-S(A(K,1))-E(K,1)
610 IF H> 1E-3THEN 660
620 PRINT K TAB(8),FNR(E(K,1)),
630 PRINT TAB(17),FNR(SQR(E(K,2))),
640 PRINT TAB(26),FNR(S(A(K,1))),
650 PRINT TAB(33),FNR(F(A(K,2)))
652 IF L>=F(A(K,2)) THEN 656
654 L=F(A(K,2))
656 V=V+E(K,2)
660 NEXT K
662
665 REM +++ DETIENE ANTES DE MOSTRAR NO CRITICAS +++
667
670 GETD$ IFD$=""THEN670
675
677 PRINT
680 PRINT TAB(10),"ACTIVIDADES NO CRITICAS"
690 PRINT TAB(10) "======================="
700 PRINT"ACT DUR D,ST CTEM  CTAR  FTEM  FTAR HOLG
```

```
710 PRINT"-----------------------------------"
720 FOR K=1 TO N
730 H=F(A(K,2))-S(A(K,1))-E(K,1)
740 IFH<=1E-3 THEN 815
750 PRINTK TAB(3) FNR(E(K,1)),
760 PRINT TAB(7) FNR(SQR(E(K,2))),
770 PRINT TAB(12) FNR(S(A(K,1)))
780 PRINT TAB(18),FNR(F(A(K,2))-E(K,1)),
790 PRINT TAB(24) FNR(S(A(K,1))+E(K,1))
800 PRINT TAB(30) FNR(F(A(K,2))),
810 PRINT TAB(35) FNR(H)
815 NEXT K
820 PRINT"-----------------------------------"
830 PRINT "TIEMPO CRITICO=" FNR(L) " +/-" FNR(SQR(V))
835
840 REM +++ DETIENE ANTES DE PROBABILIDAD NORMAL +++
845
850 GETD$ IFD$=""THEN850
860
870 REM +++ PROBABILIDAD NORMAL +++
880
940 PRINT"INGRESE TIEMPO DESEADO DE"
950 INPUT"TERMINACION (0=FIN)",D
960 IFD<=0THEN END
970 Y=(D-L)/SQR(V)
980 R= EXP(-(Y*Y)/2)/2.5066283
990 Z=Y
1000 Y=1/(1+0.33267*ABS(Y))
1010 T1=0.4361836*Y
1020 T2=0.1201676*Y
1030 T3=0.937298*Y
1040 T=1-R*(T1-T2*T2+T3*T3*T3)
1050 IF Z>=0 THEN 1070
1060 T=1-T
1070 PRINT
1080 PRINT"PROBABILIDAD DE TERMINACION EN UN"
1090 PRINT"TIEMPO DE",D," ES" FNR(T)
1100 PRINT
1110 GOTO 870
1390
2000 REM +++ SECCION DE DATA +++
2005
2010 DATA 5
2020 DATA 1,2,2,4,9
2030 DATA 2,3,3,4,9
2040 DATA 1,3,1,2,3
2050 DATA 2,4,2,4,7
2060 DATA 3,4,1,3,5
```

Conclusión

La administración de proyectos generalmente requiere de una gran habilidad personal y de una capacidad de mando significativa. Sin embargo, herramientas como las propuestas en esta serie de artículos contribuyen a facilitar la labor administrativa y constituyen una interesante aplicación de la computación a los problemas que constantemente enfrentan las empresas y los profesionales. Los métodos PERT y CPM, aunque muy simples y fáciles de usar a simple vista, requieren sin embargo de una comprensión amplia de los conceptos involucrados, y muchas veces su aplicación directa no será fácil. Sin embargo, la actual disponibilidad de microcomputadores y la facilidad que existe para escribir programas BASIC, hacen factible su utilización por un numero cada vez mayor de administradores y jefes de proyecto.

Bolsa de Empleo

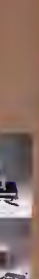
Desde este número, Microbyte pone a disposición de sus lectores un nuevo servicio tendiente a poner en contacto a aquellos que necesitan de algún profesional de la informática con quienes pueden desempeñar ese tipo de actividades.

Este servicio es absolutamente gratuito tanto para quienes buscan como para quienes ofrecen empleo y pretende ser un aporte más de Microbyte a su comunidad de lectores.

Dirija sus avisos a Bolsa de Empleo, Revista Microbyte, Merced 346, Of. "F", Santiago, los que serán publicados de acuerdo a nuestra disponibilidad de espacio.

Se ofrece Ingeniero Industrial. Experiencia en desarrollo de sistemas, análisis, programacion, estadística y transmision de datos. Llamar fono 393866.


MAI Basic Four



Programando el 6502

Jorge Cea Silva

Tal como lo prometimos, comenzamos con éste una serie de artículos en los que estudiaremos la CPU 6502 y cómo programarla. En esta serie, haremos una extensa referencia a lo ya publicado en nuestra serie sobre el Z-80 con objeto de no repetir innecesariamente y para hacer las comparaciones pertinentes. Como referencia, utilizaremos las completas Fichas Técnicas sobre el 6502 publicadas en Microbyte N° 9.

Al igual que el Z-80, este es un microprocesador de 8 bits (trabaja con palabras de 8 bits y tiene 8 líneas de datos), capaz de direccionar 64 K de memoria RAM y/o ROM (16 líneas de direcciones)

Debido a que al inicializarse un sistema con el 6502, o al resetearse, éste busca la primera instrucción a ejecutar en la dirección superior del mapa de memoria (FFFF hex), (en el caso del Z-80 en 0000), es que el Mapa de Memoria típico ubica en esta zona memoria ROM, dejando la parte inferior del mapa para memoria RAM

Bits	Registro	Bits	Registro	
		7 – 0	A	Acumulador
		7 – 0	P	De Estado
15 – 8	PCH	7 – 0	PCL	Contador de Programa
8	1	7 – 0	S	Puntero de Pila
		7 – 0	X	Registros Indices
		7 – 0	Y	Registros Indices

Registro de Estado "P"

7	6	5	4	3	2	1	0
N	V	–	B	D	I	Z	C

FIG 1 Registros del microprocesador 6502

Esta CPU tiene un diseño clásico basado en el Acumulador (A), registro principal el cual proporciona uno de los datos a la Unidad Aritmética Lógica (ALU) en toda operación aritmética y lógica, el otro dato lo obtendrá desde una localización de memoria El resultado retornará al Acumulador, desplazando el dato anterior y seteando (colocar en "1") o reseteando (colocar en "0") cuatro de los siete flags o banderas del Registro de Estado "P", estos son los flags Negativo (N), Rebalse (V), Cero (Z) y Acarreo (C), los cuales cumplen la misma función de su similar del Z-80 El resto de los 3 flags del registro de estado operan como se explica a continuación

Break (B) Bit 4 del registro de estado, el cual es automáticamente puesto en "1" por el procesador si una interrupción es causada por la instruccion BRK Este diferencia una interrupción programada (BRK) de una de Hardware (IRQ) Ninguna otra instrucción la modificará, excepto RTI y PLP (Retorno de Interrupción y Colocación del Stack en P)

Decimal (D) Bit 3 del registro de estado, el cual cuando es puesto en "1" el procesador opera en modo BCD (Binario Convertido a Decimal), y cuando está en "0" lo hace en modo binario Este flag se modificará por 4 instrucciones PLP, RTI, CLD y SED (las dos ultimas son Clear (limpiar) y Set (Colocar) el flag Decimal)

Interrupción (I) Bit 2 del registro de estado, este bit de máscara de interrupción, puede ser colocado por el programador, con las instrucciones CLI o PLP, o, por el microprocesador durante el reseteo o una interrupción Cuando está en "0" son habilitadas las interrupciones, y cuando está en "1" son deshabilitadas Las instrucciones que condicionan este bit son BRK, PLP, RTI, CLI y SEI, estas dos ultimas, Clear (limpia) y Set (coloca) el flag de interrupciones)

El registro Contador de Programa (PC) es un registro doble, de 16 bits, formado por dos registros de 8 bits PCL (bits 0 a 7), y PCH (bits 8 a 15), el cual tiene las mismas características del Z-80 y cumple las mismas funciones, resumiendo, guarda la dirección de la próxima instrucción a ser ejecutada Este es el unico registro de 16 bits de ésta CPU

El registro S almacena el Puntero de la Pila (stack), es decir la dirección más alta de la zona de memoria designada como STACK, por lo cual generalmente necesitaría 16 bits, para señalar cualquier celda de memoria como en el Z-80 Sin embargo en el 6502 el registro Stack es sólo de 8 bits más un noveno bit ubicado a la izquierda de éste (Fig 1), y que está siempre puesto en "1" Esto da como resultado que el área del Stack esté entre las direcciones 0100 y 01FF (256 y 511 en decimal ó 1 0000 0000 y 1 1111 1111 en binario) El stack deba inicializarse en la dirección 511, y puede contener hasta 255 palabras

Finalmente los últimos dos registros son X e Y, llamados Registros Indices, tienen un ancho de 6 bits. Pueden ser usados para almacenar datos, como contador y en especial como puntero índice de tablas, debido a que existen instrucciones que permiten manejarlas con gran facilidad, usando direccionamientos indexados, los cuales se verán más adelante.

Debido a que el único registro de 16 bits del 6502 es el contador de programa (PC), y no existe otro que pueda almacenar direcciones directamente, es que se hace muy importante conocer bien los modos de direccionamiento que posee esta CPU, y en especial aquellos que usan los registros índices. Cada instrucción tiene uno o más modos de direccionamiento, dependiendo de la forme y el lugar de donde obtendrá y almacenará la información.

Cuetro de estos modos de direccionamiento son igual en el Z-80 y se explicaron en la 3ª parte de Programando el Z-80 (Microbyte Nº 4), estos son IMPLICITO, INMEDIATO, RELATIVO, y ABSOLUTO, este último se explica con el nombre de Direccionamiento EXTENDIDO. En base a esto podemos confeccionar una tabla con las instrucciones equivalentes entre el 6502 y el Z-80. En el caso del Z-80 sólo algunas instrucciones utilizan direccionamiento Absoluto o Extendido, el cual utiliza 3 bytes (1 para el Código de Operación y 2 para la dirección del dato), es por ello que en algunas instrucciones mencionaremos como equivalente instrucciones con direccionamiento Registro Indirecto, 1 solo byte, el cual utiliza el registro par HL para direccionar el dato, por lo que sólo se necesitará cargar este puntero con la dirección requenda, para que después las instrucciones funcionen del mismo modo. Al final de esta tabla veremos un ejemplo.

EQUIVALENCIA DE INSTRUCCIONES DEL 6502 Y DEL Z-80

6502			Z-80
Instrucción	Direccionamiento	Operación	
ADC	Inmediato	A←(A) + n + C	ADC A, n
	Absoluto	A←(A) + (Direc.) + C	ADC A, (HL)
AND	Inmediato	A←(A) ∧ n	AND A, n
ASL	Acumulador	C ← ← A ← ← 0	SLA A
BCC	Relativo	PC←PC + e (Sólo si C = 0)	JR NC, e
BCS	Relativo	PC←PC + e (Sólo si C = 1)	JR C, e
BEQ	Relativo	PC←+ e (Sólo si Z = 1)	JRZ, e
BNE	Relativo	PC←PC + e (Sólo si Z = 0)	JR NZ, e
CLC	Implícito	C←0	AND A, A ORA, A XOR A, A (Afecta a otros flags) Si c = 1 usar CCF (Complementa el Carry)
CMP	Absoluto	A – (Mem.)	CP A, (HL)
	Inmediato	A – n	CP A, n
DEC	Absoluto	M←(M) – 1	DEC (HL)
DEX/DEY	Implícito	X←(X) – 1	DEC r
EOR	Absoluto	A←(A) ∧ (M)	XOR A, (HL)
	Inmediato	A←(A) ∧ n	XOR A, n
INC	Absoluto	M←(M) + 1	INC (HL)
INX/INY	Implícito	X←(X) + 1 Y←(Y) + 1	INC r
JMP	Absoluto	PC←(nn)	JP nn
JSR	Absoluto	STACK←(PC) + 2 PC←nn	CALL nn (SP-1)←PCH (SP-2)←PCL PC←nn
LDA	Absoluto	A←(nn)	LD A, (nn)
	Inmediato	A←n	LD A, n
LDX/LDY	Absoluto	X←(M)	LDr, (HL)
	Inmediato	X←n	LDr, n
LSR	Acumulador	0→ A →C	SRL A
	Absoluto	0→ M o (HL) →C	SRL (HL)
NOP	Implícito	Detiene la CPU 1 ciclo	NOP
ORA	Absoluto	A←A v (M)	ORA, (HL)
	Inmediato	A←A v n	OR A, n
PHA/PHP	Implícito	STACK←A/ STACK←P S←(S)-1	PUSH AF
PLA/PLP	Implícito	A←STACK/ P/←STACK S←(S) + 1	POP AF
ROL	Acumulador	A, (M) ← C	RLA
	Absoluto		RL (HL)
ROR	Acumulador	A, (M) → C	RRA
	Absoluto		RR (HL)
RTS	Implícito	PCL←(STACK) S←(S) + 1 PCH←(STACK) S←(S) + 1 PC←(PC + 1)	RET PCL←(SP) PCH←(SP + 1)
SBC	Absoluto	A←A-(M)-c̄	SBC (HL) en Z-80 el flag C no es complementado
	Inmediato	A←A-n-c̄	SBC, n
SEC	Implícito	C←1	SCF
STA	Absoluto	(nn)←A	LD (nn), A
STX/STY	Absoluto	M←(X)/M (Y)	LD (HL), r
TAX/TAY	Implícito	X←(A)/Y (A)	LDr, A
TXA/TYA	Implícito	A←(X)/A (Y)	LDA, r
TXS	Implícito	S←(X)	LD SP, rr. En Z-80 el SP es de 16 bits por lo que rr puede ser HL, IX o IY

Ejemplo: Cargar el Acumulador con el contenido de la dirección 20F0 y sumarlo al de la dirección 20F1.

a) Usando Direccionamiento Absoluto para el 6502

LDA, 20 F0	AD F0 20
CLC	18
ADC, 20F1	6D F1 20

b) Usando Direccionamiento Registro Indirecto

LDHL, 20F0	21 F0 20
LD A, (HL)	7E
INC HL	23
ADD A, (HL)	86

Como se ve en el segundo caso se ha logrado ahorrar 1 byte en este ejemplo, pero ambos deben llegar al mismo resultado.

En el próximo número veremos otros tipos de direccionamiento del 6502, así como un mayor análisis de las instrucciones que las utilizan y como es lógico ejercicios de ejemplo.

La recuperación de ECOM

Presente desde 1968 en el quehacer informativo nacional, la Empresa Nacional de Computación e Informática, ECOM, ha estado por una u otra razón en el centro de las noticias, comentarios y polémicas relacionadas con esta area

Hoy, Ecom vuelve a ser noticia y por varias razones. En primer lugar, a casi tres años de la catastrofica inundacion de sus instalaciones en Manquehue, esta institucion ha logrado recuperarse, exhibiendo utilidades operacionales y netas en el primer trimestre de este año. Al mismo tiempo, Ecom reafirma su rol como empresa netamente estatal y por ultimo, a mediados de marzo, asume María Teresa Rosende G. como Gerente General quien tuvo la gentileza de recibirnos y responder a nuestras preguntas.

Egresada en 1972 de Ingeniería Comercial en la Universidad de Concepcion, ejerció la docencia por dos años en el Departamento de Computacion de dicha Universidad. Posteriormente se traslada a Brasil donde se desempeña como analista de sistemas y Jefe de Programacion de la Universidad Estadual de Campines, donde ademas tiene la oportunidad de realizar sus estudios de postgrado en Ciencias de la Computación

Ingresa a Ecom en 1978, desempeñándose como Analista de Sistemas, Jefe de Departamento de la Gerencia de Ingeniería y Jefe del Area de Ingeniería y Explotación de Sistemas, cargo que ocupo hasta ser designada Gerente General, el 12 de marzo de este año.

Un cambio en el timón de una empresa de la envergadura de Ecom siempre genera dudas y rumores. ¿Podría explicarnos las razones de este cambio de Gerente General?

María Teresa Rosende G.

Como Ud. sabe, en abril de 1984 asumió la Gerencia General el señor Waldo Muñoz S., quien al mismo tiempo ejercía la Gerencia General de CRECIC en Concepción, empresa en la cual Corfo tenía el 67% de las acciones. A fines de 1984 Corfo licitó estas acciones, quedando desvinculada esta empresa de Ecom. En estas circunstancias el señor Muñoz no podía continuar al frente de ambas empresas y decidió continuar en CRECIC.

Sin embargo, como Ecom deseaba seguir contando con la experiencia y participacion del señor Muñoz, el directorio lo designó como su asesor técnico y a mí como Gerente General. En la práctica seguimos trabajando como antes.

¿En qué pie ha encontrado a Ecom?

En el plano económico, la situación actual de Ecom es buena y si la comparamos con un año atrás es excelente. En 1984, Ecom si bien repuntó respecto al año anterior, siempre mantuvo pérdidas, salvo en los dos últimos meses.

Este año, en los dos primeros meses ya fue posible obtener utilidades netas de alrededor de 12 millones de pesos. Esta cifra corresponde a lo que eran las estimaciones previas de lo que serían las utilidades de Ecom en todo el año. Ahora, una proyección prudente, eleva esta cifra a alrededor de 30 millones de pesos para el año. En ventas estamos bordeando los 45 millones mensuales y estamos aumentando constantemente este volumen. Recientemente se ha puesto en marcha un convenio en que Ecom se ha hecho cargo de todo el procesamiento de datos de una empresa de la envergadura de Lan Chile. Esto ha redundado en provecho para ambas empresas.

¿Como se produjo este cambio en la situación de la empresa?

Hay varias causas, aunque las más destacables son el apoyo de Corfo, que efectuó un importante aporte de capital en julio pasado ($ 800 millones), permitiendo sanear una parte importante de la deuda. Además eso dio confianza a nuestros clientes y a los potenciales clientes que no nos consideraban hasta entonces como una alternativa debido a la inestable situación de la empresa.

Otra razón, y de gran importancia, fue el extraordinario esfuerzo desplegado por todo el personal, el cual no tan sólo mantuvo a pulso el funcionamiento de la empresa, sino que además aceptó las severas medidas de restricción en los gastos.

En la época de la inundación la actitud del personal fue realmente heroica. Los que se encontraban en el lugar en el momento del desborde del río, salvaron discos y cintas hundidos en el agua y corriendo incluso riesgo por sus vidas. A continuación, se organizaron los equipos para seguir procesando la información de nuestros clien-

tes en equipos de otras empresas compatibles con nuestro equipamiento

Naturalmente, sólo teníamos acceso a esos equipos en horas de la madrugada, por lo que no era raro ver durante las horas del toque de queda a nuestros jefes de proyecto que partían con discos y formularios en sus propios autos de una empresa a otra tratando de aprovechar todas las horas disponibles de proceso

Algunas empresas fueron solidarias con nosotros y nos ayudaron muchísimo a salvar el mal rato. Otras al contrario, vieron una buena veta para sacar partido al tiempo ocioso de su equipamiento y nos cobraron en forma leonina

¿En cuánto afectó la inundación a la situación de Ecom?

Ecom antes de la inundación se había visto debilitada por una disminución en el volumen de servicios a las cajas de previsión por el proceso de traspaso a las AFP. La inundación nos tomó cuando estábamos debiles y nos dejó literalmente en la calle. Sólo en equipamiento perdimos US$ 8.5 millones, de los que por concepto de seguros recuperamos apenas el 60% y con un gran retraso. No había seguro para lucro cesante y por concepto de arrendamientos de equipo tuvimos que pagar lo que nos pedían

Sin embargo, algo que no es cuantificable es la pérdida de confianza de nuestros clientes, la que tan sólo ahora estamos volviendo a recobrar

¿Cómo ve Ud. el rol de Ecom en el país?

A partir de su creación en 1968, Ecom ha tenido como objetivo modernizar los sistemas de información para colaborar al mejor desarrollo económico y social del país. Para esto, se ha ocupado principalmente de procesar la información del sector público, capacitar técnicamente al personal de las instituciones usuarias, difundir el uso de la computación en el sector público y asesorar al gobierno en materias relativas al uso de la computación e informática

En relación a esto, creemos nuestro deber ir creando conciencia al interior del estado respecto al potencial de la informática, asesorar para un uso racional de los recursos, estandarizar los equipos y procedimientos y por último aportar en lo que podamos en el esbozo de políticas generales de informatización del país

Ese ha sido y debe seguir siendo nuestro rol. Nuestro ámbito es el sector público y es lo que justifica la existencia de una empresa como Ecom en manos del estado. Nuestro papel no está en competir con empresas del sector privado, sino en aportar cada vez más en un uso eficiente de las herramientas informáticas al interior del estado

En este tiempo en que los microcomputadores se han hecho tan populares, ¿se mantiene Ecom sólo con servicios en grandes equipos?

Definitivamente no. Pensamos que los microcomputadores deben y están jugando ya un importante rol en las instituciones públicas y Ecom debe estar presente en todos los ámbitos de la computación e informática. Por eso ya nos hemos iniciado en el desarrollo de software para microcomputadores en su utilización y en la capacitación para su uso

Deseamos convertirnos en expertos en la materia de modo que podamos asesorar, y dar servicio a todas las instituciones del Estado que lo requieran

¿Qué otros planes tienen para el futuro?

En primer lugar, continuaremos afianzando la situación de la empresa en lo económico y en lo que respecta al nivel de sus servicios. Estamos adquiriendo más equipamiento y ampliando nuestra planta de personal

A la vez, estamos desarrollando un profundo plan de capacitación intensiva del personal y diversificando nuestra gama de servicios. En este aspecto, cabe destacar el esfuerzo que ya está realizando Ecom para ampliar su labor como centro de capacitación

Indice Analítico Volumen I

A continuación, presentamos un indice no exhaustivo de los artículos publicados por Microbyte en su Volumen I, desde mayo de 1984 a abril de 1985.
El primer numaro, an negrita, corresponde a la edición, mientras el numero que sigue, es la pagina en que éste comienza.

TITULO	AUTOR	#PAG.	
TEMA educación y sociedad			
Aprender a pensar	S Papert	**1**	14
Computador en las aulas	J Kaffman	**6**	22
Computedores y enseñanza Inglés	E Edmundson	**10**	38
Computación y neg colec	P Sánchez	**11**	38
Juegos en el Atari	J Kaffman	**6**	32
Psicoware	J Kaffman	**8**	32
TEMA entrevistas			
Banco Integ de proyectos S Raggi	O Zepeda	**10**	9
Computación Educativa M Concha	J Kaffman	**11**	20
Gremio ACHEI V Celis	Microbyte	**7**	35
Informática J Mutis	Microbyte	**1**	11
Intelis Art George Cross	J López	**6**	34
Redes CTC C Haramoto	Microbyte	**8**	13
Redes Ecom P Pumarino	Microbyte	**3**	6
SONDA A Navarro	Microbyte	**8**	29
Seminario AHCIET	E Cornejo	**3**	11
Tucan V Capetillo	Microbyte	**4**	13
TEMA equipos			
Alphatronic PC	Microbyte	**4**	22
Amstrad CPC464	Microbyte	**3**	12
Apple IIC y Rana 8086/2	Microbyte	**2**	37
Computadoras del año	Microbyte	**1**	20
HP – 150	Microbyte	**3**	14
Lisa de Apple	Microbyte	**1**	12
MPF-PC	Microbyte	**6**	20
Rainbow de Digital	Microbyte	**2**	12
Tower 1632 NCR	C Faundez	**5**	12
TEMA financieros y administrativos			
Calendario	J Kaffman	**1**	38
Calendario perpetuo	A Beuchat	**8**	43
Control de Calidad	J Rojas	**10**	24
Control Proyectos (CPM)	G Beuchat	**11**	23
Evaluación de Proyectos	G Beuchat	**8**	9
Fallback	G Beuchat	**7**	16
Monto escrito	A Sarras y	**8**	18
Organización de Archivos	G Beuchat	**11**	32
Planificación	P Bill	**7**	14
Programación lineal	G Beuchat	**8**	16
Teoría de Colas	G Beuchat	**10**	20
Validación de entrada	G Beuchat	**4**	32
Validador de campos	I Rojas	**9**	19
TEMA generales			
Cómo elegir un computador	J Kaffman	**1**	25
Computadores de maletín	Microbyte	**2**	19
Ergonomía	J López	**7**	22
Industrias Lever	Microbyte	**6**	12
Inteligencia Artificial	J Kaffman	**6**	35
Mercado de los Computadores	J Kaffman	**11**	13
QuickSel Software chileno	J Kaffman	**11**	30
Sistemas Operativos	J Kaffman	**2**	25
Unix Sistema Operativo	C Faundez	**4**	6
Viene el Videodisco	L Kaffman	**5**	32
TEMA hardware			
El diseño de computadoras	L Kaffman	**2**	30
Ficha 6502	J Aravena	**9**	25
Ficha 8086	J Aravena	**10**	17
Ficha ASCII-EBDIC-CPU 6748	J Aravena	**11**	15
Fierros #1	V Capetillo	**2**	14
Fierros #2	V Capetillo	**3**	17
TEMA juegos			
Aerocaza submarino	J Periale	**10**	44
Ahorcado	Microbyte	**7**	39
Ataque submarino	J Vargas	**5**	42
Bandera chilena	M Campodonic	**7**	42
Bioritmo	M Costa	**11**	49
Cuadrados mágicos	J Kaffman	**10**	47
Cuadrado par	J Kaffman	**11**	46
Creación de sonidos Atari	Microbyte	**2**	36
Dibujar	P Muller	**3**	35
Equilibrista	J Alvarez	**4**	35
Estrella perfecta	G Concha	**9**	44
Gráficos y animación	Microbyte	**4**	36
Gráficos y sonidos #1	G Mery	**5**	36
Gráficos y sonidos #2	G Mery	**6**	39
Gráficos y sonidos #3	G Mery	**9**	38
Juego del gato	G Gómez	**8**	38
Misiles	Microbyte	**8**	40
Morse	A y J Aravena	**6**	40
Palitroques	Microbyte	**8**	41
Recursión Torre de Hanoi	C Contreras	**11**	43
Rompecabezas	Microcentro	**2**	33
Septiembre (Bandera)	J Arancibia	**5**	42
Slalom	G Mery	**7**	32
Sopa de letras	Microcentro	**5**	40
Toque y fama	Microbyte	**1**	33
Torero	Microbyte	**8**	41
TEMA lenguajes			
DbaseII Base de datos	H Miranda	**9**	21
Sistema Link de Burroughs	C Teixido	**8**	32
Software integrado	L. Kaffman	**3**	22
TEMA noticias IBM-PC compatibles			
Compatibilidad IBM	J Kaffman	**4**	8
Noticias IBM-PC compatibles	Microbyte	**2**	6
Noticias IBM-PC compatibles	Microbyte	**1**	5
TEMA programación			
BvB#1 Instruccion Print	J Kaffman	**1**	22
BvB#2 Diagramas de flujo	J Kaffman	**2**	22
BvB#3 Criptografía	J Kaffman	**3**	24
BvB#4 Juegos de azar	J Kaffman	**4**	24
BvB#5 Redondeo	J Kaffman	**5**	24
BvB#6 Subrutinas	J Kaffman	**6**	24
BvB#7 Arreglos	J Kaffman	**7**	24
BvB#8 Funciones	J Kaffman	**8**	24

TITULO	AUTOR	#PAG
BvB#9 Gráficos	J Kaffman	9 27
BvB#10 Numeros romanos	J Kaffman	10 28
Estructurar vs optimizar	M Vaisman	6 29
Glosario de términos #1	M Bernoff	3 26
Glosario de términos #2	M Bernoff	4 16
Programando el Z-80 #1	J Cea	2 16
Programando el Z-80 #2	J Cea	3 19
Programando el Z-80 #3	J Cea	4 14
Programando el Z-80 #4	J Cea	5 14
Programando el Z-80 #5	J Cea	6 10
Programando el Z-80 #6	J Cea	7 10
Programando el Z-80 #7	J Cea	8 14
Programando el Z-80 #8	J Cea	9 14
Seudoestructuras BASIC #1	R Puvogel	4 18
Seudoestructuras BASIC #2	R Puvogel	5 19
Uso Sist Operativo CP M (1)	J Aravena	10 13
Uso Sist Operativo CP M (2)	J Aravena	11 40

TEMA Software de sistema

Call y Poke en Apple	H Saavedra	4 38
Cambio de color de fondo	G Contreras	7 41
DLIST	E Ahumada	11 51
Editor Sharp pc 1500	A Schaedler	6 31
Editor de pentalla apple	H Saavedra	3 41
Ensamblador 6502 #1	E Ahumada	7 32
Ensamblador 6502 #2	E Ahumada	8 26
F list Atari	A Beuchat	8 44
Flash	Microbyte	8 42
Lectura del taclado Timex	C Contreras	4 34
Lectura y RUN simultáneas	Microbyte	7 41
Lenguaje Máquina en Atari	Microbyte	3 40
Listados Atari protegidos	Microbyte	6 42
Mejores listados F list	G Beuchat	6 37
Peeks y pokes Atari	Microbyte	1 35
Proteger programas Atari	Microbyte	2 35

TITULO	AUTOR	#PAG.
Programe su ensamblador	E Ahumeda	10 30
Qué lee el Sinclair	A Aravena	3 38
Rutina de edición Atari	Microbyte	9 42
Subrutinas Atari	Microbyte	3 39
Titulo de lectura forzada	Microbyte	9 44
Toque personal con Apple	Microbyte	2 38

TEMA Software técnico

Conversión bases numéricas	H Miranda	10 45
Cronómetro	J Alegria	8 42
Ecuaciones Simultáneas	I Azocar	6 41
Factores	J Kaffman	10 43
Graficador de pulsos	C Contreras	4 28
Más problemas imposibles	C Contreras	3 9
Matrices	L. Manriquez	9 40
Problemas imposibles	J Kaffman	1 9
Raices de Ecuaciones	M Benavente	11 17
Regresion lineal	G Beuchat	3 8
Simulación	G Beuchat	5 10
Simulación continua	O Vesa	7 18
Sorting	G Beuchat	2 31
Técnica de Montecarlo	J Kaffman	2 9
Teoria de grafos	O Vega	5 27

TEMA Telemática

Canales de Comunicaciones	C Contreras	4 26
Comentarios	J Aravena	1 30
Comunicación entre Timex	C Contreras	3 37
Dialog bases de datos	C Gual	8 29
Fibras ópticas	R Abarca	8 32
La Red	C Contraras	3 32
Misión imposible	C Contreras	5 37
Protocolos	P Pumarino	6 14
Red local o telemática	C Contreras	3 30
Trans de datos en Chile	E Bonnelont	4 10

Círculos de Calidad

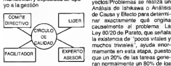
Pedro Sánchez A.
Miembro de la Asoc. Chilena de Relaciones Industriales y Miembro del Sindicato ECOM.

Tormenta de Ideas, Análisis de Ishikawa, Brainstorming son conceptos que irrumpen con la llamada Administración participativa que busca ligar los recursos humanos de toda la pirámide organizacional, en la busqueda creativa de ideas y procedimientos que mejoren el posicionamiento competitivo de la empresa.

Indudablemente esta filosofía de integración tiene una brillante concreción en los denominados "Círculos de Calidad", creados en Japón en los inicios de la década del 60, y que, posteriormente partan en las principales empresas de los Estados Unidos.

Se puede definir sucintamente un Círculo de Calidad como un grupo de personas de una misma área de actividades que se organiza en forma absolutamente voluntaria para analizar problemas de su competencia laboral en forma sistemática y coordinada, buscando soluciones eficientes e implementándolas en conjunto con la dirección de la empresa. El principal objetivo de un Círculo de Calidad es el aumento de la productividad y la disminución de costos.

El esquema está basado en una estructura muy simple de organización informal a la cual se le reconocen canales de comunicación y capacidad de apoyo a la gestión.



Comité Directivo es el encargado de plantear los objetivos terminales y del entrenamiento global de los participantes.

Líder es el encargado de dirigir, controlar y motivar a todos los miembros del Círculo, además determina las responsabilidades individuales de cada miembro.

Facilitador es la persona encargada de coordinar la alta gerencia con todos y cada uno de los Círculos de Calidad existentes en la empresa. Un facilitador puede administrar part time hasta 7 círculos.

Miembros son los recursos humanos que trabajan directamente en la resolución de problemas de acuerdo al esquema de desarrollo.

Toda esta organización, previa en labores de entrenamiento y capacitación para crear un clima organizacional sano y participativo en el cual además exista el respaldo y compromiso de los ejecutivos, demora aproximadamente 6 meses.

Durante las primeras reuniones deben dirigirse las sesiones con una excelente dinámica de grupos que permita integrar a cada miembro de un círculo en forma vital, sin inhibiciones, aplicando el concepto de Tormenta de Ideas, cada persona menciona todo lo que cree digno de tratar. Posteriormente se analizan los planteamientos y se jerarquizan las prioridades de acuerdo con la contribución de cada idea al desarrollo productivo de la organización.

Obtenido un listado de Proyectos/Problemas se realiza un Análisis de Ishikawa o Análisis de Causa y Efecto para determinar exactamente qué origina causalmente el problema. La Ley 80/20 de Pareto, que señala la existencia de 'pocos vitales y muchos triviales', ayuda enormemente en esta etapa, puesto que un 20% de las tareas generan normalmente un 80% de los resultados.

Una poderosa herramienta analítica es el concepto de las 4 M (maquinarias, materiales, métodos y mano de obra) inserto en el Diagrama de Ishikawa.

M M EFECTO M M

Dicho diagrama permite reconstruir las causas que posibilitan el efecto negativo y tomar las acciones correctivas pertinentes.

Los Círculos de Calidad tienen un desarrollo potencial en las empresas de gestión moderna y fundamentalmente en empresas o departamentos de Computación, dadas las características de homogeneidad profesional, trabajo en equipo y la necesidad de ganar y generar experiencias válidas en esta joven ingeniería que es la Computación.

En el extranjero las principales empresas japonesas tienen todas Círculos de Calidad, al igual que en Estados Unidos; en nuestro país empresas como IBM, AFP Santa María, Banco de Chile, tienen implementados dichos círculos y están en un período de enriquecimiento de experiencias y técnicas; DATA GENERAL y ECOM evalúan las posibilidades.

La disciplina de la administración se ve enriquecida permanentemente por los aportes de las empresas que desarrollan tecnologías de punta. Buscar formas inteligentes de utilizar dichos métodos de administración que posibiliten una mayor integración de los estamentos laborales aparece como un desafío para el área de Relaciones Industriales y como una necesidad imperiosa para el exigente sector de la Computación e Informática en el cual el factor fundamental es el potencial humano.

Modelamiento de datos

Guillermo Beuchat

La técnica conocida como "modelamiento de datos" o "modelamiento de la información", ha sido presentada en diversas ocasiones en la literatura por autores especializados (MARTIN & FINKELSTEIN, M. FLAVIN, E. F. CODD, etc.). Dichos autores proponen un enfoque analítico y estructurado para efectuar un análisis global y corporativo con el fin de identificar los datos relevantes a una organización y las relaciones entre ellos.

Como técnica para el análisis de sistemas, el modelamiento de datos consiste básicamente en la identificación de "entidades" y las relaciones lógicas existentes dentro de un sistema real. Aunque no es de amplia difusión, se usa para el diseño de grandes bases de datos que integran la información de toda una organización, y para el diseño de SIA's en diversos ámbitos. Se ha demostrado que la técnica es muy útil aun dentro de ambientes tradicionales de desarrollo de sistemas computacionales, pues permite identificar claramente toda la información que debe almacenar un sistema para cumplir su función, y es posible obtener un "modelo de datos" que refleja fielmente el problema de la vida real.

El propósito de este artículo es mostrar algunos aspectos teóricos y prácticos de esta metodología de diseño de sistemas y bases de datos, que sin duda es una de las herramientas más útiles que se han desarrollado en el ámbito del análisis de sistemas.

Componentes de un modelo de datos

Según M. FLAVIN, un modelo de datos está compuesto por cinco componentes básicos: entidades o sujetos de datos, relaciones, operaciones, atributos o elementos de información, y restricciones.

– Entidades:
son los "entes lógicos" de información o núcleos primitivos de datos de un sistema. Por ejemplo, personas, cosas, documentos, lugares, etc. M. Flavin define incluso un super-conjunto de entidades llamado "sujeto de datos", pero su aplicación se restringe sólo al estudio global de organizaciones completas (modelo de datos corporativo).

– Relaciones:
son las asociaciones lógicas existentes entre dos o más entidades de un modelo de datos. Estas asociaciones tienen un nombre propio y diversas características que analizaremos más adelante.

– Operaciones:
una operación es una acción que modifica el estado del sistema que se está modelando, es decir, corresponde a una transacción efectuada sobre los datos del sistema.

Existen dos clases de operaciones: standard y definidos por el analista. Operaciones standard típicas son del tipo CREAR, MODIFICAR, ELIMINAR y las definidas por el analista pueden ser del tipo DESPACHAR-MERCADERIA, EMITIR-CHEQUE, etc.

– Atributos:
son los datos elementales que describen a una entidad o a una relación. Por ejemplo, algunos atributos de la entidad EMPLEADO podrían ser RUT, NOMBRE, CODIGO, CARGO, SUELDO, etc.

– Restricciones:
las restricciones corresponden a limitaciones impuestas al modelo de datos en todos sus aspectos (contenido de datos, definición de los mismos, etc.) y que dependen de las políticas de diseño de sistemas y bases de datos de la organización.

Cada uno de los cinco componentes lógicos que plantea Flavin puede a su vez describirse mediante seis características que lo individualizan:

– Nombre:
es necesario nombrar cada componente preferiblemente con una palabra o un conjunto de palabras que tengan relación nemotécnica con el componente.

– Definición:
cada componente se define de acuerdo al rol que juega dentro del modelo de datos.

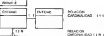
– Contenido:
el conjunto de atributos que están asociados a una entidad o relación se denominan "contenido de datos" del componente. Por otra parte, el atributo que identifica en forma unívoca a cada entidad se denomina "identificador" de cada ocurrencia de la entidad. Un ejemplo típico es el atributo RUT para una entidad PERSONA.

– Estructura:
la estructura de datos de un componente está asociada con el número de ocurrencias de cada atributo dentro de una entidad o relación. Por ejemplo, la entidad DEUDOR tendrá asociados los atributos NOMBRE y RUT como atributos no-repetitivos y los atributos LETRA y MONTO como atributos repetitivos dentro de la misma entidad.

– Operaciones:
las operaciones permitidas sobre un componente

deben definirse previamente. Por ejemplo, una entidad puede ser eliminada, se pueden agregar ocurrencias, etc.

– Dependencia:
es necesario definir las relaciones existentes dentro de los atributos asociados a cada componente del modelo. Por ejemplo, si CODIGO-ZONA = 00 entonces NOMBRE-ZONA = REGION METROPOLITANA.







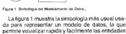


Figura 1. Simbología del Modelamiento de Datos.

La figura 1 muestra la simbología más usual usada para representar un modelo de datos, la que permite visualizar rápida y fácilmente las entidades y las relaciones entre ellas. Es necesario definir el concepto de "cardinalidad", tal como se muestra en la figura.

La "cardinalidad" corresponde al numero de ocurrencias de cada entidad asociada a una relación. Por ejemplo, la entidad EMPLEADO tendrá asociada una sola ocurrencia de la entidad CONTRATO-DE-TRABAJO, mientras que podrá tener múltiples ocurrencias de la entidad CARGA-FAMILIAR.

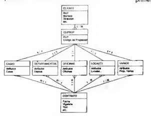


Fig. 2. Sistema de Corretaje de Propiedades.

La figura 2 muestra el modelo de datos general de un sistema de corretaje de propiedades, que fue desarrollado en Chile en 1984 usando la metodología de modelamiento de datos. A partir de este modelo, se implementó directamente el sistema en el lenguaje de bases de datos relacional d BASE II, lo que permitió obtener un sistema de consulta interactiva a toda la cartera de propiedades de un corredor. Este modelo se presenta como ejemplo del resultado de aplicar el modelamiento de datos a un problema de la vida real.

Normalización de un modelo

El análisis formal de los datos de un sistema usando el concepto de entidades y relaciones produce normalmente un modelo generalizado que llamaremos modelo conceptual del sistema. Sin embargo, este modelo generalmente tendrá lo que podríamos llamar "fallas estructurales", tales como datos redundantes, repetitivos o relaciones de cardinalidad N : N.

Para obtener el llamado "modelo canónico" o modelo primitivo del sistema, es necesario aplicar un proceso de tres etapas llamado NORMALIZACION al modelo, cuyo objetivo es obtener un modelo de datos que:

- sea independiente de cualquier SIA o aplicación particular que se desee diseñar
- esté compuesto solamente de relaciones simples y entidades.

A continuación se describe el proceso de normalización, que debería culminar con la obtención de un modelo de datos canónico.

e) Primera Forma Normal

La primera normalización consiste en eliminar de todas las entidades del modelo conceptual, aquellos atributos repetitivos que son en sí entidades. Por ejemplo, la entidad DEUDOR, con n ocurrencias de los atributos LETRA, MONTO y FECHA-VENCIMIENTO, podría separarse en dos entidades DEUDOR y LETRA, relacionadas entre sí mediante una relación 1 : N. La figura 3 ilustra esta primera normalización.

ENTIDAD PRIMITIVA

DEUDOR
RUT
Nombre
Dirección
Letra 1
Monto 1
Fecha venc. 1
Letra 2
Monto 2
Fecha venc. 2

PRIMERA FORMA NORMAL

DEUDOR
RUT
Nombre
Dirección

1 : N

LETRA
Monto
Fecha venc.

Fig. 3. Normalización Primera Forma Normal.

b) Segunda Forma Normal

La segunda normalización consiste en solucionar el problema de las relaciones de cardinalidad **N : N**, que no permiten "navegar" por el modelo para tener acceso a los datos. Por ello, se construyen entidades intermedias llamadas NUBS, que transforman una relación **N : N** en dos relaciones **1 :N**, que representan la misma relación. Generalmente, este tipo de entidades no contiene atributos, aunque es posible encontrar NUBS cuyo contenido de datos corresponde a coeficientes técnicos entre las 2 entidades. La figura 4 muestra la segunda normalización sobre las entidades ALUMNO y CURSO. Un alumno puede tener asociados varios cursos, y a la vez, un curso puede tener asociados varios alumnos. Por ello, se construye el NUB ALUMNO/CURSO, que soluciona el conflicto.

RELACION PRIMITIVA

ALUMNO — N:N — CURSO

SEGUNDA FORMA NORMAL

ALUMNO

1:N

Alum./Cur NUB — N:1 — CURSO

Fig. 4. Normalización Segunda Forma Normal

ENTIDAD PRIMITIVA

DEUDOR

RUT
Nombre
Dirección
Teléfono
Nombre cliente
Dirección cliente

TERCERA FORMA NORMAL

DEUDOR

RUT
Nombre
Dirección
Teléfono

N:1

CLIENTE

Dirección cliente
Nombre cliente

Fig. 5. Normalización Tercera Forma Normal

c) Tercera Forma Normal

La tercera normalización consiste básicamente en la separación de atributos relacionados entre sí para formar una nueva entidad. Por ejemplo, la entidad DEUDOR podría tener asociados los atributos NOMBRE-CLIENTE y DIRECCION-CLIENTE. Sin embargo, estos atributos se repiten para muchas ocurrencias de la entidad DEUDOR, y corresponden más bien a una entidad separada llamada CLIENTE. La normalización produce entonces el modelo que se muestra en la figura 5.

Otra falla estructural que es necesario resolver en un modelo conceptual es la existencia de trayectorias cerradas entre las entidades, que constituyen una redundancia. Estas trayectorias deben "cortarse" de tal forma que se resuelva el conflicto, manteniendo la posibilidad de "viajar" de una entidad a otra a través de las relaciones que queden. La figura 6 ilustra un caso típico de esta anormalidad en el modelo, usando las entidades ALUMNO, CURSO y NOTAS.

RELACION PRIMITIVA

ALUMNO — NUB — CURSO

NOTAS

TERCERA FORMA NORMAL

ALUMNO — NUB — CURSO

NOTAS

Fig. 6. Normalización Trayectoria Redundante

ENFOQUE DE MARTIN & FINKELSTEIN

Este enfoque orientado al análisis global de grandes grupos de datos o áreas de la organización, provee una secuencia ordenada de pasos a seguir para obtener el modelo de datos de un sistema, tal como lo hemos descrito hasta aquí. Es importante notar que la secuencia que se presenta permite obtener el modelo de datos, pero no constituye una manera de diseñar un SIA cualquiera usando esos datos. Dado el modelo, el analista podrá aplicar incluso técnicas tradicionales de diseño (en que las entidades pasan a ser archivos) o bien técnicas integradoras mediante el uso de Sistemas de Administración de Bases de Datos y software de cuarta generación que permita implementar directamente el modelo en una configuración computacional.

El enfoque de Martin & Finkelstein consta de dos fases, llamadas Análisis de la Información y Síntesis de Datos. La primera fase se puede estructurar en seis pasos específicos.

1. Identificación inicial de los datos

Este proceso considera los objetivos y ele-

mentos de política estable de la organización, a fin de determinar los datos fundamentales sin los cuales la organización no existiría. Permite generar un modelo de datos inicial, en que se ha identificado y agrupado todos los atributos relevantes a cada tipo de dato, se han definido los atributos identificadores y se han identificado grupos repetitivos.

2. Modelamiento de datos por área funcional

Se consideran las diferentes funciones de la organización (mercado, servicios, productos, etc.) a suficiente nivel de detalle. El análisis debe hacerse con previsión para futuros datos, y deben definirse explícitamente todos los atributos asociados a cada entidad del modelo de función. Además, las relaciones entre entidades deben definirse a partir de las dependencias funcionales del sistema.

3. Extensión de los datos a objetivos unitarios

Definidos y jerarquizados los objetivos de la organización, se definen los datos necesarios para la toma de decisiones a diferentes niveles y se definen las medidas de efectividad del cumplimiento de los objetivos planteados. Con ello, se extiende el modelo de datos hasta cumplir esos objetivos.

4. Revisión de datos actuales

Se consideran los datos reales, y la manera en que están documentados y relacionados entre sí, con el fin de eliminar inconsistencias. Es decir, se consideran los datos "como son" y no como "deberían ser".

5. Integración de modelos funcionales

Se combinan los modelos de datos parciales obtenidos en la etapa 2, generando un conjunto global de atributos y entidades. Se efectúa una normalización para eliminar redundancias e inconsistencias. Finalmente, se obtiene un modelo integrado de datos.

6. Resolución de conflictos de los datos

Es necesario definir los procedimientos de manipulación de datos comunes a muchos usuarios dentro de la organización, a fin de no causar conflictos en la administración de la base de datos.

Por otra parte, el proceso de Síntesis de Datos puede definirse mediante cuatro etapas, que se realizan a continuación de haber obtenido el modelo de datos de la fase anterior. El objetivo de estas cuatro etapas es la obtención del modelo conceptual canónico del sistema, apto para el futuro procesamiento de datos mediante la implementación del modelo en una adecuada combinación de hardware y software. Las cuatro etapas de esta fase son:

1. Análisis de los datos

Se hace un análisis a nivel de atributos y datos elementales, a fin de obtener el máximo de detalle posible en el contenido de datos del modelo.

2. Síntesis canónica

Se normaliza totalmente el modelo, obteniendo un modelo canónico o de tercera forma normal, desprovisto de redundancias, datos repetidos o repetitivos, etc.

3. Resolución de conflictos

Se resuelven los conflictos de definición de datos, tales como nombres diferentes para el mismo dato, etc.

4. Análisis de estabilidad

Se analiza la sensibilidad del modelo frente a cambios en los datos para hacer frente a necesidades futuras.

Una vez obtenido el modelo de datos de un sistema mediante la aplicación de esta técnica, se tiene una clara visión de la factibilidad de diseñar un SIA cualquiera dentro del modelo. De hecho, el diseño lógico tradicional puede acortarse significativamente si se cuenta con un modelo de datos global de la organización, y el diseño físico, puede hacerse directamente a partir del modelo si se cuenta con el software apropiado. Sin embargo, la mayoría de paquetes de bases de datos disponibles presentan restricciones, especialmente en lo que se refiere a la implementación de las relaciones entre las entidades. Ello obliga muchas veces a "corromper" el diseño introduciendo datos redundantes o entidades adicionales con el fin de superar las restricciones.

Existe un último punto importante que tratar, que tiene que ver con la utilidad del modelo conceptual para su posterior implementación. En efecto, una base de datos sólo será útil en la medida en que sea factible solucionar con ella los más variados requerimientos de información. Para extraer los datos se realiza un proceso llamado "navegación", que consiste en recorrer las entidades y relaciones del modelo de acuerdo a una trayectoria pre-establecida.

Una vez constituido el modelo conceptual canónico es necesario, por lo tanto, obtener el llamado "modelo lógico", que contempla todas las trayectorias de navegación para satisfacer los requerimientos de información a la base de datos. Muchas veces, este modelo lógico no será óptimo desde el punto de vista de la tercera forma normal, pero sí será posible cumplir los requerimientos en forma eficiente.

La construcción del modelo lógico constituye entonces una etapa de transición para el uso de un modelo conceptual en la implementación de un SIA específico, y su descripción escapa al alcance de este artículo.

Para aquellos lectores que deseen profundizar más estos temas, se recomienda la siguiente bibliografía:

1. Sistemas de Administración de Bases de Datos. Oscar Barros. Ed. Universitaria.
2. Fundamental Concepts of Information Modeling. Matt Flavin. Yourdon Press, 1981.
3. Extending the Database Relational Model to Capture Data Meaning. E.F. Codd. ACM Transactions on Database Systems. Dic. 1979.

OPENFILE

Cartas del lector

COMPILADOR

Señor Director

Me resulta muy agradable dirigirme a Ud. para formularle algunas consultas y hacer algunas aclaraciones.

En primer lugar, me gustaría referirme a la petición del señor R. A. Rozas (enero, pág. 36) que se declara aburrido del Basic y pregunta por compiladores existentes. Aparte de los que ustedes ya mencionaron, existe un programa compilador para el ZX-81 llamado ZX-Compiler, y que es un compilador Basic (me imagino que el lector lo aburre la lentitud y no el Basic en sí).

También deseo señalar un error que se deslizó en la misma página, en el listado del programa Assembler, en él ustedes van incrementando el acumulador partiendo de uno, entonces lo que el computador imprime es el CHR$ de 1 a 5 y no los números que debería.

Finalmente, me gustaría hacerles cuatro peticiones.

Por favor expliquen cómo hacer los fantásticos gráficos tridimensionales que aparecen en el número de enero, página 35, ya que me gustaría reproducirlos pero no me queda claro cómo.

Si fuera posible hagan algunos concursos de programación como el que hicieron en la sección "bienvenidos al Basic", ya que son un gran incentivo para todos los aficionados a la programación.

Traten, en lo posible, de publicar más sobre Hardware y lenguaje de máquina para los Sinclair ya que hay poca literatura sobre esto, en cambio programas de juego se encuentran por montones en todas partes.

Desearía intercambiar correspondencia con personas de todas las edades sobre los siguientes temas: ZX-81, Spectrum, Timex 1000, Timex 2068, Sinclair QL, Casio FX-702-P, Procesador Z80, Procesador 68000 etc.

Saluda atentamente,

Julio Rojas G.
Murphy 263
Viña del Mar

P.D. Les adjunto dos listados con explicaciones e instrucciones.

Le agradecemos sus aclaraciones y sugerencias, las que esperamos satisfacer próximamente. Los listados que nos envía serán publicados en la sección Sinclair.

TRADUCCION ATARI-SINCLAIR

Señores Microbyte

Referente a la sección consultas técnicas me dirijo a ustedes para lo siguiente. Con respecto al programa "Operando con matrices", página Nº 40 del número del mes de enero de 1985, quisiera preguntarles de qué manera podría modificar aquel programa para poder ejecutarlo en un computador ATARI 800XL, ya que las órdenes SLOW, FAST e INKEY$ no sé por cuáles sustituirlas. Agradecería mucho que me pudieran solucionar este problema.

Sin otro particular, y esperando un éxito cada vez mayor de su revista, se despide muy atentamente de ustedes,

Jorge Urrutia C.
Pedro Lira 1346
Santiago

En principio, las instrucciones SLOW y FAST son sólo útiles en un ZX-81, por lo que no habría que traducirlas al querer tipear el programa en un Atari. Sencillamente, omítalas.

La utilidad de estas instrucciones en un Sinclair es que en modo SLOW el computador, a la vez que va haciendo sus cálculos internos, también debe controlar todo lo que aparece en la pantalla, lo que lo hace más lento. En modo FAST, el computador se olvida de la pantalla, hace sus cálculos a toda velocidad y luego despliega los resultados en la pantalla.

El Atari es más rápido, por lo que no necesita ese tipo de instrucciones.

La instrucción INKEY$ en el Sinclair es similar a lo que hacemos en el Atari cuando queremos conocer la última tecla que se ha presionado. Por ejemplo, si tipeamos en un Sinclair

10 PRINT INKEY$
20 GOTO 10

y echamos a correr este miniprograma, en la pantalla aparecerá siempre la última tecla que presionamos.

Para hacer lo mismo en el Atari debemos escribir

10 PRINT PEEK(764)
20 GOTO 10

CORRESPONDENCIA ATARI

Ciro Calderón W. de Calama, nos ha escrito pidiendo sea publicada su solución para intercambiar programas e información para el Atari. Los interesados le escriban a Uno Sur 860 Villa Exótica, Calama.

Además, nos ha enviado este pequeño, luminoso y ruidoso programa para quienes quieran experimentar.

```
1 REM >>>PARA MICROBYTE
2 REM ***********************
3 REM * [illegible]          *
4 REM * POR CIRO CALDERON W. *
5 REM *    MARZO DE 1985     *
6 REM ***********************
10 GRAPHICS 8+16
20 FOR P=1 TO 100
30 COLOR 1
40 C=INT(RND(0)*3):D=INT(RND(0)*255)
50 A=INT(RND(0)*319)+1:B=INT(RND(0)*191)
+1
60 DRAWTO A,B:SOUND C,D,10,8
70 NEXT P:GOTO 10
```

SORT POR INDICES

Sr. Director:

Tengo a bien hacerles llegar un listado de programa a modo de colaborar para vuestra publicación.

La función del programa consiste en la creación de un archivo de índices para un archivo de transacciones contables usando el algoritmo de Shell para el sort, siendo útil su aplicación para cualquier microcomputador.

Las ventajas que ha reportado este procedimiento de sort interno son:

- clasificación por más de una clave (cuatro en este caso)
- ahorro de espacio en disquete, el archivo generado no ocupa más de 15 sectores
- rapidez de proceso, este programa se corre en un Olivetti M20 y para 1 200 registros demora aproximadamente 6 minutos (el utilitario Olisort ocupa cerca de una hora para igual volumen además del intercambio de discos para los archivos de trabajo y de salida)
- las claves de selección pueden modificarse fácilmente según las necesidades del usuario o crearse más de un archivo de índices por otros campos elegidos sin alterar el archivo de entrada y manteniendo todo en un mismo disco.

Descripción General

líneas	
160 a 260	almacenamiento del string que agrupa los campos seleccionados en A$ (i) y su posición en Z% (i). Cada campo es convertido previamente en su expresión alfanumérica
270	la variable FIN conserva el número de registros a clasificar
300 a 380	clasificación del array A$ e intercambio de índices en Z%. En 330 se comparan las variables hasta un largo de 26 que es el máx. de de la suma de las cuatro variables
400 a 450	Grabación en el file PUNTEROS del contenido del array Z%

El acceso posterior al file "ARCHIVO" se efectúa leyendo secuencialmente el file PUNTEROS y el valor obtenido será la key para accesar el movimiento contable, el fin del archivo se detecta por el valor cero de dicha key.

Esperando que la presente contribución sea de interés para su publicación, les saluda Atte.

Ernesto Hódar Rodríguez
Cinco Norte 835
Viña del Mar

```
10 REM PROGRAMA CLASIFICADOR   ORDENA ARCHIVO CONTABLE POR
20 REM                                        - CUENTA
30 REM                                          RUT
40 REM                                        - TIPO DOCUMENTO
50 REM                                        - NUM  DOCUMENTO
60 REM                  --- ON ASCENDING KEY ---
70 REM               *** ERNESTO HODAR RODRIGUEZ ***
80 CLS
90 DEFINT I,J,N
100 DIM A$(1300)
110 DIM Z%(1300)
120 OPEN "R",1,"0.ARCHIVO",128
130 FIELD 1 4 AS C1$ 5 AS C2$ 8 AS C3$ 2 AS C4$ 4 AS C5$ 20 AS C6$ 2 AS C7$ 30
          AS C8$ 9 AS C9$ 2 AS C$ 5 AS C$(1) 5 AS C$(2) 5 AS C$(3) 27 AS NOMBRE$
140 PRINT "          LEYENDO ARCHIVO CONTABLE "
150 CURSOR (10,10) PRINT " REGISTRO NUMERO ----) "
160 FOR J=1 TO 9999
170 GET 1 J
180 CURSOR(35,10) PRINT J
190 IF CVD(C3$)=0 THEN 270
200 CUENTA%=CVS(C1$)    CUENTA$=STR$(CUENTA%)
210 TIPO%=CVS(C$(1))    TIPO$=STR$(TIPO%)
220 NUMERO$ = STR$(CVS(C$(2)))
230 RUT$=MID$(C9$ 1,9)
240 RETA$=CUENTA$ + RUT$ + TIPO$ + NUMERO$
250 A$(J)=RETA$   Z%(J)=J
260 NEXT J
270 FIN%=J - 1
280 CLOSE 1
290 CLS PRINT " INICIO DE SORT "
300 DD=2^INT(LOG(FIN%)/LOG(2)) - 1
310 FOR I=1 TO FIN% - DD
320                FOR J%=1 TO I STEP -DD
330                 IF MID$(A$(J%),1,26) <= MID$(A$(J% + DD),1,26) THEN 370
340                      SWAP Z%(J%)   Z%(J% + DD)
350                      SWAP A$(J%)   A$(J% + DD)
360                NEXT J%
370 NEXT I
380 DD=INT(DD/2)     IF DD>0 THEN 310
390 PRINT " *** FIN SORT - GRABANDO ARCHIVO PUNTEROS *** "
400 OPEN "R" 2 "0 PUNTEROS" 2
410 FIELD 2 2 AS LUGAR$
420 FOR J=1 TO FIN%
430 LSET LUGAR$=MKI$(Z%(J))

440 PUT 2
450 NEXT J
460 CLOSE 2
470 PRINT "*** FIN GRABACION PUNTEROS ***"
480 END
```

VIDEO INVERSO

Señor director:

En su revista N° 10, correspondiente al mes de marzo, se comenta la implementación de un video inverso, por software, para el ZX-81 y a lo cual usted sugieren un programa que realiza esta propiedad.

Sinclair Chile, por medio de su Servicio Técnico, ofrece a los usuarios de microcomputadores la disposición de un servicio para la conversión de televisores transistorizados a monitor y con video normal o inverso manipulado por interruptores.

Este sistema, por ser con señal de video, entrega una imagen completamente estabilizada y con video normal se puede conectar a cualquier tipo de computador que tenga señal de video.

Para ello, el interesado nos deberá hacer llegar a nuestro Departamento Servicio Técnico el televisor y el computador para su adaptación.

Es conveniente hacer presente que el usuario no pierde el uso normal en su televisor.

Les damos la presente como alternativa a su idea original.

Ing. Jorge Cea S.
Jefe Servicio Técnico
Sinclair Chile Ltda.

CORRECCIONES

Sr Director

Mis pnmeras palabras, como ya es costumbre para Ud , son de felicitaciones por una producción de tan alta calidad, lamentablemente no todo puede ser flores ya que nuevamente aparecen errores de digitación en la edición 10 en la linea 65 del programa Aerocaze submarino falta el dibujo del submarino y en el programa de Video inverso la linea 210 debe decir 210 LET Y = PEEK 16397

Además envio este mismo programa en lenguaje de máquina y el cargador hexadecimal en el cual se puede ingresar más de un codigo a la vez porque el programa de carga aparecido en el curso del señor Cea postenormente corregido, no funciona

Seguire colaborando con su revista y si existen interesados en intercambio tengo un Timex 1000 16K, un VIC-20 16 k con Vicmon y un gran surtido de programas para ambos computadores

Le saluda muy atentamente

Gustavo Vargas T
Gratel Ltda
Agustinas 657 Of 11, Stgo

Programa de carge hexadecimal. 1

```
 10 REM
 20 LET N   16514
 30 INPUT U$
 40 IF U$    S  THEN STOP
 50 IF LEN U$ - 2 THEN GOTO 30
 60 LET L   CODE U$(1)-28
 70 LET L   16 ı L ı CODE  U$(2)
    -28
 80 IF L   255 THEN GOTO 30
 90 POKE N L
100 PRINT U$(1 TO 2)
110 LET U$   U$(3 TO)
120 LET N   N ı 1
130 GOTO 40
```

DLIST

Señor Director

Por medio de la presente deseo agradecerle la gentileza que usted y su aquipo editonal me han brindado al publicar mi articulo acerca del programa DLIST Sin embargo deseo hacer algunos alcances sobre el mismo, para beneficio de los lectores que deseen usar el programa programa

- Este programa fue desarrollado onginalmente pera un VIC con 21 kbytes de memoria y diskettera Sin embargo, reconociendo el hecho de que muy pocos VIC-20 disponen de este periférico, el listado que acompaña al articulo es de una versión para Commodore C-64
- Aquellos dueños de VIC-20 que no hayan podido hacer funcionar el programa pueden tomar contacto conmigo para ofrecerles una copia correcta
- Para quienes posean un C-64 y deseen entonces tipear el listado, hago una "Fe de Errata , pues la linea 200 debiera decir
  200 IF NA$(P7)() THEN F7=P8 C7=2 GOSUB 940 LOAD NA$(P7),8

Finalmente, deseo agradecer a todos aquellos amigos que me han escrito a raiz de mis articulos antenores dándome apoyo e intercambiando experiencias

Eduardo Ahumada Mazuranich
Dominica 103, Depto 34
Santiago

NOS ESCRIBE

Juan Román T., profesor de Limache (Casilla 167), nos ha enviado un programa de utilidad para su actividad y además nos consulta por métodos para obtener alta resolución en un ZX-81 y mayor información sobre PEEKs y POKEs

Al respecto, por el momento no nos es posible publicar un programa de alta resolución, pero si podemos informarle que existen programas en el mercado que lo hacen Esperamos sin embargo poder publicar un programa similar en un futuro próximo

En relación a los PEEKs además de lo ya publicado en el propio manual de ZX en los capitulos 26 al 28 no es mucho lo que se puede abundar Sin embargo, estamos preparando una lista de subrutinas llamables de la ROM las que facilitaran enormemente la programacion en lenguaje de máquina

El programa que nos envia lo hemos puesto en pauta para una proxima edición

Cristián R Abarzua F. de San Martin 768 Dp 32 de Quilpué solicita se incluyan en la revista programas para el TRS-80

Nosotros estamos preparando material, pero si otro lector desea colaborar bienvenido sea

Carlos Castro, de Angol 127 – Concepcion, desea intercambiar programas con otros usuarios de Atari

Programa Video inverso 2

	Assembler	Hex	Comentario
	LD HL,(400C)	2A 0C 40	Carga en HL la dirección anterior al comienzo del archivo de pantalla
	LD B 14h	06 14	Carga el valor 20 en B
X1	INC HL	23	Suma 1 a HL
	LD A,(HL)	7E	Deja en el acumulador el valor de la dirección contenida por HL
	CP 76h	FE 76	Si el acumulador tiene el valor 118
	JP Z 4094	CA 94 40	salta a X2
	ADD A 80	C6 80	Invierte el caracter sumando 128 al acumulador
	LD (HL) A	77	Coloca el caracter invertido en pantalla
	JP 4087h	C3 87 40	Salta a X1
X2	DEC B	05	Resta 1 a B
	JP NZ 4087	C2 87 40	Si B no es cero salta a X1
	RET	C9	Regresa al BASIC

Cargue con el programa 1 los datos HEX adecimales y ejecute con RAND USR 16514


SMITH-CORONA
La respuesta americana en impresoras.
Exíjalas en los mejores negocios de computación.


EPSON
RX-80 Personal Printer
CONECTABLE
A TODO
COMPUTADOR